GCA (S5) 04A
Exam. Code: DWD

Data Warehousing and Data Mining

SEMESTER V

BACHELOR OF COMPUTER APPLICATION

Block 2



KRISHNA KANTA HANDIQUI STATE OPEN UNIVERSITY

Subject Experts

Prof. Anjana Kakati Mahanta, Gauhati University

Prof. (Retd.) Pranhari Talukdar, Gauhati University

Dr. Jyotiprokash Goswami, Associate Professor, Assam Engineering College

Course Co-ordinators

Dr. Tapashi Kashyap Das, Assistant Professor, KKHSOU

Ms. Sruti Sruba Bharali, Assistant Professor, KKHSOU

SLM Preparation Team

UNITS	CONTRIBUTORS
9, 10, 14, & 15	Dr. Naba Jyoti Sarmah, Nalbari Commerce College
11 & 12	Ms. Daisy Kalita, USTM
13	Mr. Biswajit Das, Cotton University

Editorial Team

Content : Mr. Gautam Chakrabarty, NERIM

Language : Prof. (Retd.) Robin Goswami, Cotton College

Structure, Format & Graphics: Ms. Sruti Sruba Bharali, KKHSOU

June, 2019 ISBN: 978-93-89123-31-9

This Self Learning Material (SLM) of the Krishna Kanta Handiqui State University is made available under a Creative Commons Attribution-Non Commercial-ShareAlike4.0 License (International): http.//creativecommons.org/licenses/by-nc-sa/4.0.

Printed and published by Registrar on behalf of the Krishna Kanta Handiqui State Open University.

Headquarters : Patgaon, Rani Gate, Guwahati-781 017

City Office : Housefed Complex, Dispur, Guwahati-781 006; Web: www.kkhsou.in

The university acknowledges with thanks the financial support provided by the **Distance Education Bureau, UGC** for the preparation of this study material.

BACHELOR OF COMPUTER APPLICATION

Data Warehousing and Data Mining

Block-2

CONTENTS

		Pages
UNIT 9:	**Association Rule Mining** Association Rule Mining, Market Basket Analysis, Apriori Algorithm, Mining Multilevel Association Rules, From Association Mining to Correlation Analysis	**141-159**
UNIT 10:	**Classification** Introduction to Classification, Classification by Decision Tree, Attribute Selection Measure, Covering Rules	**160-175**
UNIT 11:	**Prediction** Introduction to Prediction Techniques, Statistical (Bayesian) Classification, Bayesian Networks, Instance-based Methods (Nearest Neighbor)	**176-190**
UNIT 12:	**Evaluation** Training and Testing, Evaluating Accuracy of a Classifier, Cross-Validation, Combining Multiple Models (Bootstrap, Boosting, Bagging)	**191-204**
UNIT 13:	**Clustering** Introduction to Clustering, Classification of Various Clustering Algorithms, Partitioning Methods– k-means, Density Based DB-SCAN, Hierarchical Methods– Agglomerative and Divisive Clustering	**205-214**
UNIT 14:	**Introduction to Web Mining** Web Mining Introduction, Web Content Mining, Web Structure Mining, Web Usage Mining, Text Mining, Unstructured Text, Episode Rule Discovery for Text	**215-233**
UNIT 15:	**Introduction to Spatial and Temporal Data Mining** Temporal Data Mining, Temporal Association Rules, Sequence Mining, Spatial Mining, Spatial Mining Tasks, Spatial Clustering	**234-260**

BLOCK INTRODUCTION

This is the second block of the course ***Data Warehousing and Data Mining***. After completing this block, learners will be able to understand the concepts of data mining like association rule mining, classification, prediction, evaluation and clustering. In addition to these, new data mining topics like web mining, spatial mining and temporal data mining are also introduced to the learners.

This block comprises the following seven units:

Unit 9 gives us an introduction to association rule mining. Market based analysis, apriori algorithm, mining multilevel association rules are some of the topics covered in this unit.

Unit 10 introduces us to classification. Classification by decision tree, covering rules and attribute selection measures are some of the topics discussed in this unit.

Unit 11 deals with prediction. The different prediction techniques, statistical classification, bayesian networks, instance based methods like nearest neighbour are discussed in this unit.

Unit 12 deals with evaluation. Different topics like training and testing, evaluating accuracy of classifier, cross validation etc are dicussed in this unit.

Unit 13 describes clustering method. Topics like different clustering algorithms, partitioning methods- k-means, Density based DB-SCAN, Hierarchical methods- agglomerative and divisive clustering are discussed in this unit.

Unit 14 introduces us to web mining. Topics like web content mining, web structure mining, web usage mining, text mining etc are discussed in this unit.

Unit 15 gives us an introduction to spatial and temporal data mining. Topics like temporal data mining, temporal association rules, sequence mining, spatial mining, spatial clustering etc are covered in detail in this unit.

Each unit of this block includes some along-side boxes to help you know some of the difficult, unseen terms. Some “EXERCISES” have been included to help you apply your own thoughts. You may find some boxes marked with: “LET US KNOW”. These boxes will provide you with some additional interesting and relevant information. Again, you will get “CHECK YOUR PROGRESS” questions. These have been designed for you to self-check your progress of study. It will be helpful for you if you solve the problems put in these boxes immediately after you go through the sections of the units and then match your answers with “ANSWERS TO CHECK YOUR PROGRESS” given at the end of each unit.

UNIT 9: ASSOCIATION RULE MINING

UNIT STRUCTURE

9.1 Learning Objectives

9.2 Introduction

9.3 Association Rule Mining

9.4 Market Basket Analysis

9.4.1 Some Useful Definitions

9.5 Apriori Algorithm

9.6 Multilevel Association Rules

9.6.1 Approaches to Mining Multilevel Association Rules

9.6.2 Checking for Redundant Multilevel Association Rules

9.7 From Association Mining to Correlation Analysis

9.8 Let Us Sum Up

9.9 Further Reading

9.10 Answers to Check Your Progress

9.11 Model Questions

9.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- define association rule
- describe market basket analysis
- describe Apriori algorithm for association rule mining
- define multilevel association rule mining
- describe how correlation analysis is performed.

9.2 INTRODUCTION

In the previous unit, in the first block, we have discussed topics like attribute generalization and attribute relevance. We have also learned about different statistical measures in the previous unit. In this unit, we will learn about association rule mining and market basket analysis. We will also learn how apriori algorithm works. We will also cover the different multilevel

association rules. The unit also covers how correlation analysis is performed from association mining. In the next unit, we will explore the concept of classification in detail.

9.3 ASSOCIATION RULE MINING

Association rule mining is a data mining technique that discovers the probability of the co-occurrence of items in a collection of data. The relationships between co-occurring items are expressed as association rules. Association rules are often used to analyse sales transactions. For example, it might be noted that customers who buy 'bread' at the grocery store often buy 'butter' at the same time. In fact, association analysis might find that 85% of the buyers that include 'bread' in their buying list also include 'butter'. This relationship could be formulated as the following rule.

'Bread' implies 'butter' with 85% confidence

This application of association modelling is called market-basket analysis. It is valuable for direct marketing, sales promotions, and for discovering business trends. Market-basket analysis can also be used effectively for store layout, catalogue design and cross-sell.

Association modelling has important applications in other domains as well. For example, in e-commerce applications, association rules may be used for web page personalization. An association model might find that a user who visits KKHSOU website and MHRD website is likely to also visit UGC website in 70% times in the same session. Based on this rule, a dynamic link could be created for users who are likely to be interested in the 3rdpage. The association rule could be expressed as follows.

KKHSOU website and MHRD website imply UGC website with 70% confidence.

9.4 MARKET BASKET ANALYSIS

Market Basket Analysis is the most typical example of association mining. Customers Invoices (Bill) are recorded in all the supermarkets. This database, known as the "market basket" database, consists of a large number

of records on past transactions. A single record lists all the items bought by a customer in one sale. Knowing which items are sold together with which set of items gives these shops the freedom to adjust the store layout and the store catalogue to place the optimally concerning one another.

Example 9.1: Database with 4 items and 5 transactions

Table 9.1: Market Basket Dataset

Transaction ID	Milk	Bread	Butter	Beer
1	1	1	0	0
2	0	0	1	0
3	0	0	0	1
4	1	1	1	0
5	0	1	0	0

9.4.1 Some Useful Definitions

Before we discuss more about Association Rule Mining we need to be familiar with few terms that are going to be used in association rule mining

ITEMSET: An item set I is a set of n items represented as I = {i_1, i_2, i_3, …, ..., ..., i_n}. In the above example (Table 9.1) milk, bread, butter and beer are the items in the itemset I.

TRANSACTION: A transaction 't' consists of a transaction_id and a set of items which are present in that transaction. In the table 9.1 first transaction (Transaction_id 1) consist of only milk and butter.

DATASET: A dataset **D** is a set of 'n' transactions, where each transaction 't' contains a non-empty set of items represented as **D** = {t_1, t_2, t_3, …, ..., ..., t_n}

SUPPORT: Support refers to the frequency of an itemset. It is measured by the number of transactions in which the itemset appears. It is referred as Supp(X) where X is the set of items. In the Table 9.1, the support of {bread} is $\frac{3}{5}$, or 60%. Itemsets can also contain multiple items. For instance, the support of {milk, bread, butter} is $\frac{1}{5}$, or 20%.

CONFIDENCE: Confidence refers to how likely item Y is purchased when item X is purchased, expressed as {X → Y}. This is measured by the number of transactions with item X, in which item Y also appears.

Confidence is defined as Conf (X → Y) = $\frac{\text{Supp}(X \cup Y)}{\text{Supp}(X) \times \text{Supp}(Y)}$.

In Table 9.1, the confidence of {milk → bread} is 100%

i.e., Supp (milk U bread) / Supp (milk) =>(40% / 40%) and confidence of {milk, bread → butter} is 50%.

LIFT: The lift value is a measure of importance of a rule. The lift value of an association rule is the ratio of the confidence of the rule and the expected confidence of the rule. The expected confidence of a rule is defined as the product of the support values of the rule body and the rule head divided by the support of the rule body. Lift is defined as Lift (X → Y) = $\frac{\text{Supp}(X \cup Y)}{\text{Supp}(X) \times \text{Supp}(Y)}$

In Table 9.1, the lift of {milk → bread} is 10/6 = 1.6

i.e., Supp (milk U bread) / [Supp (milk) x Supp (bread)]

$$= \frac{\frac{2}{5}}{\frac{2}{5} \times \frac{3}{5}} = \frac{\frac{2}{5}}{\frac{6}{25}} = \frac{2}{5} \times \frac{25}{6} = \frac{10}{6} = 1.6$$

A lift value greater than 1 means that item Y is *likely* to be bought if item X is bought, while a value less than 1 means that item Y is *unlikely* to be bought if item X is bought.

9.5 APRIORI ALGORITHM

Apriori is a frequent item set mining and association rule learning algorithm over transactional databases. It proceeds by identifying the frequent individual items in the database and extending them to larger item sets. The frequent item sets determined by Apriori can be used to determine association rules which highlight general trends in the transactional database. The Apriori algorithm was proposed by Agrawal and Srikant in 1994. Apriori

is designed to operate on databases containing transactions. Given a threshold T, the Apriori algorithm identifies the item sets which are subsets of at least T transactions in the database.

Apriori uses a “bottom up” approach, where frequent subsets are combined to generate the candidates for next iteration as one item at a time. The algorithm terminates when no further successful extensions are possible.

Apriori uses breadth-first search and a Hash tree structure to count candidate item sets efficiently. It generates candidate item sets of length‘k’ from item sets of length ‘k-1’. Then it prunes the candidates which have an infrequent sub pattern in the pruning step by using the downward closure property. According to downward closure property “All Subsets of a frequent itemsets must be frequent”. After that, it scans the transaction database to determine frequent item sets among the candidates.

The algorithm is given below for a transaction database ‘T’, and a support threshold of ‘Th’. At each step, the algorithm is assumed to generate the candidate sets from the large item sets of the preceding level, heeding the downward closure lemma.

The Apriori algorithm is a collection of three different procedures, the main apriori algorithm, candidate generation and pruning. Bellow all three algorithms are discussed with details.

Algorithm 9.1: Apriori algorithm

Input: Transaction database and a user-defined minimum supportTh

Output: All frequent itemsets

Step 1: $L_0 := \varnothing$; $k := 1$;

Step 2: $C_1 := \{\{i\} \mid i \in I\}$

Step 3: Answer := $\varnothing$

Step 4: **while** $C_k \neq \varnothing$

Step 5: read database and count supports for C_k

Step 6: L_k:= {frequent itemsets in C_k}

Step 7: C_{k+1} := *Apriori-gen* (L_k)

Step 8: $k := k + 1$

Step 9: **end while**

Step 10: Answer := Answer $\cup$ L_k

Step 11: **return** Answer

Apriori-gen procedure used above in the Apriori is combination of two procedures, *candidate generation*and a *prune* procedure. The *candidate generation* procedure combines two frequent *k*-itemsets, which have the same (*k*–1)-prefix, to generate a (k+1) itemset as a new preliminary candidate. Following the *candidate generation* procedure, the *prune* procedure is used to remove infrequent itemsets from the preliminary candidate set all itemsets '*c*' such that some *k*-subset of '*c*' is not a frequent itemset.

Algorithm 9.2: The *candidate generation* procedure of the Apriori-gen algorithm

Input: L_k, the set containing frequent item sets found in pass *k.*

Output: Preliminary candidate set C_{k+1}.

Step 1: **for** i **from** 1 **to** $|L_k-1|$

Step 2: **for** j **from** i + 1 **to** $|L_k|$

Step 3: **if** L_k.itemset$_i$ and L_k.itemset$_j$ have the same (k–1)-prefix

Step 4: $C_{k+1} := C_{k+1} \cup \{ L_k.itemset_i \cup L_k.itemset_j\}$

Step 5: **break**

Algorithm 9.3: The *prune* procedure of the Apriori-gen algorithm

Input: Preliminary candidate set C_{k+1} generated from the *join* procedure above.

Output: final candidate set C_{k+1} which does not contain any infrequent itemset.

Step 1: **for** all itemsets*c* in C_{k+1}

Step 2: **for** all *k*-subsets *s* of *c*

Step 3: **if** $s \notin L_k$

Step 4: **delete** *c* from C_{k+1}

Example 9.2: Consider the following market basket dataset and minimum threshold value as 3 (60%). If we apply the Apriori Algorithm then we get the following results.

Transaction ID	Items Bought
T1	{Mango, Onion, Nintendo, Key-chain, Eggs, Yo-yo}
T2	{Doll, Onion, Nintendo, Key-chain, Eggs, Yo-yo}
T3	{Mango, Apple, Key-chain, Eggs}
T4	{Mango, Umbrella, Corn, Key-chain, Yo-yo}
T5	{Corn, Onion, Key-chain, Ice-cream, Eggs}

For simplify the example we use M- for Mango, O- for Onion, N- for Nintendo etc. After simplifying the dataset looks as follows

Transaction ID	Items Bought
T1	{M, O, N, K, E, Y }
T2	{D, O, N, K, E, Y }
T3	{M, A, K, E}
T4	{M, U, C, K, Y }
T5	{C, O, K, I, E}

The above dataset can also be converted to binary dataset as follows:

Transaction ID	M	O	N	K	E	Y	D	A	U	C	I
T1	1	1	1	1	1	1	0	0	0	0	0
T2	0	1	1	1	1	1	1	0	0	0	0
T3	1	0	0	1	1	0	0	1	0	0	0
T4	1	0	0	1	0	1	0	0	1	1	0
T5	0	1	0	1	1	0	0	0	0	1	1

Using the same procedure we can find the frequent itemsets from binary datasets. In experiments we use only the binary datasets since it is easy to calculate frequencies from binary datasets.

Each pass of Apriori algorithm is consist of four steps: Frequency counting, Frequent itemset selection, Candidate generation and Pruning. In the following workout all these four steps are discussed in detail.

Step 1: *Frequency Counting*– Count the number of transactions in which each item occurs, this is also known as frequency counting for one itemsets.

Item	Number of Transactions
M	3
O	3
N	2
K	5
E	4
Y	3
D	1
A	1
U	1
C	2
I	1

Step 2: *Frequent Item Selection*– Depending on the user provided minimum threshold value frequent items of each pass is selected. Here our predefined threshold value is 3. So frequent one items will be those items which have frequency 3 or above. Following are the frequent one items

Item	Number of Transactions
M	3
O	3
K	5
E	4
Y	3

Step 3: *Candidate Generation*– from the frequent one item sets the candidates for the next pass will be generated by following the *Candidate Generation* algorithm. Here all the frequent one items are combined to generate the probable two item candidate sets. In the following table all such two item candidate sets are listed

Item pairs
MO
MK
ME
MY

OK
OE
OY
KE
KY
EY

Step 4: *Pruning*– Since two item candidate sets are generated only from the one item frequent sets so pruning does not prune any two item candidate sets. So same candidate sets are passed to the second pass for frequency counting.

Step 5: *Frequency Counting*– In the second pass, the frequency counting is performed for two item candidate sets. Here, we count how many times each pair of items are bought together. For example, MO is bought together only in the transaction T1{M,O,N,K,E,Y}

While MK are bought together 3 times in T1{M,O,N,K,E,Y}, T3{M,A,K,E} AND T4{M,U,C,K,Y}

After counting the frequencies of all such item pairs we get,

Item Pairs	Number of Transactions
MO	1
MK	3
ME	2
MY	2
OK	3
OE	3
OY	2
KE	4
KY	3
EY	2

Step 6: *Frequent Item Selection*– After observing the frequencies in the above table the frequent two itemsets are found as follows:

Item Pairs	Number of transactions
MK	3
OK	3
OE	3
KE	4
KY	3

These are the pairs of items frequently bought together.

Step 7: *Candidate Generation*– To generate the candidates for next pass (i.e., three itemsets) we have look at the following conditions.

The three item candidates will be generated only if prefix of both the frequent itemsets are same and differ by only the last item for example

OK & OE are frequent itemsets and OK & OE differ by K & E(only the last items), this gives OKE (Common + differ items or in other words {O,K} $\cup$ {O,E})

Similarly from KE and KY, we get, KEY

After observing all the frequent two items we get candidates for next pass are

Item Pairs
OKE
KEY

Step 8: *Pruning*– In the pruning step all the subsets of a candidates are checked whether they are frequent or not, if not then that item is removed from the candidate sets.

Here, subsets of OKE are {OK},{KE},{OE} and all three subsets are frequent so this three item sets are not pruned.

For KEY the subsets are {KE},{EY}and{KY} here {KE} and{KY} are frequent but {EY} is not frequent so the candidate KEY is removed from candidate list.

The final candidate list after pruning is as follows

Item Pairs
OKE

Step 9: *Frequency Counting*– In the third pass the frequency counting is performed for three item candidate sets.

After counting the frequencies of all such item pairs we get

Item Pairs	Number of transactions
OKE	3

Step 6: *Frequent Item Selection*– After observing the frequencies in the above table the frequent three itemsets are found as follows

Item Pairs	Number of transactions
OKE	3

Since only one frequent item pairs left so no more candidate sets can be generated hence the algorithm stops here.



CHECK YOUR PROGRESS

Q.1: What does Apriori algorithm do?

a) It mines all frequent patterns through pruning rules with lesser support

b) It mines all frequent patterns through pruning rules with higher support

c)

d)

Q.2: What do you mean by support (A)?

a) Total number of transactions containing A

b) Total Number of transactions not containing A

c) Number of transactions containing A / Total number of transactions

d) Number of transactions not containing A / Total number of transactions

Q.3: How do you calculate Confidence (A $\rightarrow$ B)?

a) Support(A $\cap$ B) / Support (A)

b) Support(A $\cap$ B) / Support (B)

c) Support(A $\cup$ B) / Support (A)

d) Support(A $\cup$ B) / Support (B)

Q.4: Which of the following is direct application of frequent itemset mining?

a) Social Network Analysis

b) Market Basket Analysis

c) Outlier Detection

d) Intrusion Detection

9.6 MULTILEVEL ASSOCIATION RULES

For many applications,it is difficult to find strong associations among data items at low or primitive levels of abstraction due to the diversity of data in multidimensional space. Strong associations discovered at high concept levels that might represent common sense knowledge. However, what may represent common sense to one user may seem novel to another. Therefore, data mining systems should provide capabilities to mine association rules at multiple levels of abstraction and traverse easily among different abstraction spaces.

Association rules generated from mining data at multiple abstraction levels are called multiple-level or multilevel association rules. Multilevel association rules can be mined efficiently using concept hierarchies under a support-confidence framework.

9.6.1 Approaches to Mining Multilevel Association Rules

In general a top-down strategy is employed,where counts are accumulated for the calculation of frequent itemsets at each concept level, starting at the highest concept level and working towards the lower, more specific concept levels, until no more frequent itemsets can be found. That is, when all frequent itemsets at all the concept levels are mined and no more frequent itemsets can be generated the algorithm stops. For each level,any algorithm for discovering frequent itemsets may be used, such as Apriori or its variations by using following two approches.

- **Using uniform minimum support for all levels:** The same minimum support threshold is used when mining at each level

of abstraction. When a uniform minimum support threshold is used, the search procedure is simplified. The method is also simple in that users are required to specify only one minimum support threshold. An optimization technique can be adopted, based on the knowledge that an ancestor is a superset of its decedents;the search avoids examining itemsets containing any item whose ancestors do not have minimum support.

The uniform support approach, however, has some drawbacks. It is unlikely that items at lower abstraction levels will occur as frequently as those at higher abstraction levels. If the minimum support threshold is set too high, it could miss some meaningful associations occurring at low abstraction levels. If the threshold is set too low, it may generate many uninteresting associations occurring at high abstraction levels. This leads to the next approach.

- **Using reduced minimum support at lower levels:** Each level of abstraction has its own minimum support threshold. The threshold value is smaller for lower abstraction level, and higher for higher level of abstraction. For mining multiple-level associations with reduced support,there are a number of alternative search strategies such as Level-by-Level independent where a breadth first search is performed without considering the knowledge of previous frequent itemsets. Here, each data is checked regardless of whether or not its parent is frequent or not. Next approach is level-cross-filtering by one item where an item at the i^{th} level is examined if and only if its parent node at the $(i-1)^{th}$ level is frequent. If a node is frequent then its children are examined; otherwise, its descendants are not considered in the next search. Next one is level-cross filtering by -K-itemset where instead of one item in the i^{th} level k-itemsets are examined, if and only if the corresponding parent k-itemset at the $(i-1)^{th}$ level is frequent then only the candidate k-itemsets are considered at i^{th} level.

- **Using item or group-based minimum support:** Because of users or experts often have the insight knowledge about a particular item groups, which one is more important than others, it is sometimes more beneficial to set up a user-specific item group-based minimal support thresholds when mining multilevel rules. For example, a user could set up the minimum support thresholds based on product of interest such as by setting particularly low support thresholds for frying pan and rice cooker to pay particular attention on the association patterns containing these two items.

  For mining patterns with mixed items from groups with different support thresh-olds, usually the lowest support threshold among all the participating groups is taken as the support threshold for mining. This will avoid filtering out valuable patterns containing items from the group with the lowest support threshold. In the meantime, the minimal support threshold for each individual group should be kept to avoid generating uninteresting association rules.

  For mining these types of datasets with different groups of data efficient methods can be developed or extend the exsisting methods to mine these type of datasets.

9.6.2 Checking for Redundant Multilevel Association Rules

A serious side effect of mining multilevel association rules is that it generates many redundant rules across all the multiple abstraction levels. Also concepts of hierarchy are useful in data mining since they permit the discovery of knowledge at different levels of abstraction, such as multilevel association rules. However, when multilevel association rules are mined,some of the rules found will be redundant due to the ancestors at the previous level. So special precisions are needed to be taken up to reduce generation of such redundant rules.

9.7 ASSOCIATION MINING AND CORRELATION ANALYSIS

For filtering uninteresting association rules support and confidence are not sufficient. To overcome this correlation measure is used along with support and confidence for mining association rules.

Correlation rule is used measure not only from the support and confidence but also it uses the correlation between itemsets. There are many different correlation measures from which we can choose for rule mining.

Lift is a simple correlation measure. The definition of lift is discussed in section 9.4.1. If the value of lift is less than 1, then the occurrence of A is negatively correlated with the occurrence of B, meaning that the occurrence of one likely leads to the absence of the other one. If the resulting value is greater than 1, then A and B are positively correlated, which means that the occurrence of one implies the occurrence of the other. If the resulting value is equal to 1, then A and B are independent and there is no correlation between them. Lift is usefull to filter out misleading “strong” associations rules of the form $A \Rightarrow B$.

The second correlation measure that we study is the x^2 measure. In (chi-square) test also correlation relationship between two attributes, A and B can be discovered. To compute the x^2 value, we take the squared difference between the observed and expected value for a pair in the contingency table, divided by the expected value. This amount is summed for all such pairs in the contingency table.

Suppose A has c distinct values, namely a_1, a_2, ..., ..., ..., a_c. B has r distinct values, namely b_1, b_2, ..., ..., ..., b_r. The data tuples described by A and B can be shown as a contingency table, with the c values of A making up the columns and the rvalues of B making up the rows. Let (A_i, B_j) denote the joint event that attribute A takes on value a_i and attribute B takes on value b_j. Each and every possible (A_i, B_j) joint event has its own cell in the table. The x^2 value (also known as the Pearson x^2 statistic) is computed as:

$$x^2 = \sum_{i=1}^{c} \sum_{j=1}^{r} \frac{(o_{ij} - e_{ij})^2}{e_{ij}}$$

where o_{ij} is the observed frequency (i.e., actual count) of the joint event (A_i, B_j) and e_{ij} is the expected frequency of (A_i, B_j) which can be computed as:

$$e_{ij} = \frac{count(A = a_i) \times count(B = b_j)}{n}$$

Here, n is the number of data tuples, *count* ($A = a_i$) is the number of tuples having valuea_i for A, and *count* ($B = b_j$) is the number of tuples having value b_j for B. The sum in the equation for computing x^2 is computed over all of the *r* × *c* cells. Note that the cells that contribute the most to the value are those for which the actual count is very different from the expected.

The statistic tests the hypothesis that A and B are independent, that is, there is nocorrelation between them. The test is based on significance level, with *(r* – 1*)* × *(c* – 1)degrees of freedom.



CHECK YOUR PROGRESS

Q.5: Why is correlation analysis important?

a) To make apriori memory efficient

b) To weed out uninteresting frequent itemsets

c) To find large number of interesting itemsets

d) To restrict the number of database iterations

Q.6: Lift value 0 means?

a) A is negatively correlated with the occurrence of B

b) A and B are positively correlated

c) A and B are not related to each other

d) None of the above

Q.7: For multilevel association rule mining

a) Threshold is not used

b) Minimum threshold is computed automatically

c) Multiple minimum threshold is used

d) None of the above

Q.8: x^2 (chi-square) test is used to–

a) check the validity of a rule

b) compute the frequency count

c) compute the validity of an itemset

d) check whether an itemset is in the transaction or not



9.8 LET US SUM UP

- Association rule mining is a data mining technique that discovers the probability of the co-occurrence of items in a collection of data.
- Association rules are often used to analyse sales transactions.
- One of the applications of association modelling is market-basket analysis. Which is valuable for direct marketing, sales promotions, and for discovering business trends.
- Association modelling has important applications in other domains as well. For example, in e-commerce applications, association rules may be used for Web page personalization.
- Apriori is a frequent item set mining and association rule learning algorithm over transactional databases.
- Apriori proceeds by identifying the frequent individual items in the database and extending them to larger item sets.
- The frequent item sets determined by Apriori can be used to determine association rules which highlight general trends in the transactional database.
- Apriori uses a "bottom up" approach, where frequent subsets are combined to generate the candidates for next iteration as one item at a time.
- The Apriori algorithm is a collection of three different procedures, the main apriori algorithm, candidate generation and pruning.
- Association rules generated from mining data at multiple abstraction levels are called multiple-level or multilevel association rules. Multilevel association rules can be mined efficiently using concept hierarchies under a support-confidence framework.

- For mining multilevel association rule mining one of these three methods are used, using uniform minimum support for all levels, using reduced minimum support at lower levels or using item or group-based minimum support
- A serious side effect of mining multilevel association rules is that it generates many redundant rules across all the multiple abstraction levels.
- For filtering uninteresting association rules support and confidence are not sufficient. To overcome this correlation measure is used along with support and confidence for mining association rules.
- Correlation rule is used measure not only from the support and confidence but also it uses the correlation between itemsets.
- Lift and x^2 (chi-square) test are the two common approach for correlation measure.



9.9 FURTHER READING

1) Aggarwal, C. C. (2015). *Data Mining: The Textbook*. Springer.
2) Han, J., Pei, J. & Kamber, M. (2011). *Data Mining: Concepts and Techniques*. Elsevier.
3) Tan, P. N. (2018). *Introduction to Data Mining*. Pearson Education India.



9.10 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q. No. 1: (a) **Ans. to Q. No. 2:** (c)
Ans. to Q. No. 3: (a) **Ans. to Q. No. 4:** (b)
Ans. to Q. No. 5: (b) **Ans. to Q. No. 6:** (c)
Ans. to Q. No. 7: (c) **Ans. to Q. No. 8:** (a)



9.11 MODEL QUESTIONS

Q.1: What is support and confidence? Explain with example.

Q.2: What is association rule? Explain briefly.

Q.3: Consider the database in table 9.1 and suppose min support threshold is 2. Use the Apriori algorithm to generate all the frequent itemsets in the dataset.

Q.4: Discuss the different methods used for multilevel association rule mining?

Q.5: What is Lift? Explain with example.

Q.6: Explain the x^2 (chi-square) test? What is the use of it in rule mining?

Q.7: Explain in detail:

a) Market Basket analysis

b) Multilevel association rule mining

c) Correlation analysis

*** ***** ***

UNIT 10: CLASSIFICATION

UNIT STRUCTURE

10.1 Learning Objectives

10.2 Introduction

10.3 Classification

10.4 Decision Tree

10.4.1 Decision Tree Algorithms

10.5 Attribute Selection Measure

10.5.1 Entropy and Information Gain

10.5.2 Alternative Measures for Selecting Attributes

10.6 Drawbacks of Using Decision Trees

10.7 Let Us Sum Up

10.8 Further Reading

10.9 Answers to Check Your Progress

10.10 Model Questions

10.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- define classification
- describe the basic techniques of data classification
- explain what is decision tree
- explain how to build decision tree classifiers
- describe the different measures for attribute selection for decision tree.

10.2 INTRODUCTION

In the previous unit, we have learned about association rule mining, market basket analysis and different multilevel association rules. We have also learned about how apriori algorithm works and how correlation analysis is performed from association mining. In this unit, we will discuss about Classification and Decision trees. Different attribute selection measures like entropy and information gain are also discussed in this unit. In the next unit, we will explore the concept of prediction in detail.

10.3 CLASSIFICATION

Classification is a data analysis technique used to classify large amount of data into different predefined classes. This technique is used to predict category of data and assign class labels. For example, we can build a classification model to categorize network traffic data into different classes such as attack or normal data. Such analysis provides us a better understanding of the large amount of data. Many classification methods have been proposed by researchers in machine learning, pattern recognition, and statistics. Most algorithms are memory resident, typically assuming a small data size. Recent data mining research has developed scalable classification and prediction techniques capable of handling large amounts of disk-resident data. Classification has numerous applications, including fraud detection, target marketing, performance prediction, manufacturing, and medical diagnosis etc.

A network administrator is interested to know whether the incoming request is “attack” or “normal data” or a bank loan officer needs analysis which loan applicants are “safe” and which are “risky” for the bank. A medical researcher wants to analyze MRI image for identifying “tumours”. In each of these examples, the data are classified according to some predefined criteria, where a classifier is constructed to predict class (category) of the data, such as “attack” or “normal” in network administrator or “safe” or “risky” for the loan application data or “tumour” or “not a tumour” for the MRI data. These categories can be represented by discrete values, where the ordering among values has no meaning.

Regression analysis is a statistical methodology that is most often used for numeric prediction; hence classification and regression are used synonymously. Some other methods for numeric prediction also exist. Classification and numeric prediction are the two major types of prediction problems.

Data classification is a two-step process, *learning step* and *classification step*. In the learning step the classification model is constructed and in the classification step class labels are predicted by the classification model.

In the *learning step*, a classifier is built to classify data into a predetermined set of data classes or concepts. This phase is also known as training phase, where a classification algorithm builds the classifier by analyzing the training dataset. The training dataset is also similar to normal datasets except that each tuple in the training dataset has a predefined class label. Using this predefined class labels learning phase builds the classifier by using any of the classification algorithms. The class label is discrete-valued and unordered. The class label indicates the data category of the tuple.

Since the class label of each training tuple is provided in the training dataset classification problem is termed as *supervised learning*. It contrasts clustering as an unsupervised learning method where no information about the class label and number of class is provided beforehand. For example, in a network traffic data if we did not have the class label as which data is normal or which one is attack for the training data, we may use clustering to try to groups the data based on the clustering algorithm. Clustering is discussed in UNIT 13.

In the first step of classification(learning step), data or tuples of training data sets are mapped with the associated class labels. In the second step, these mappings are used to predict the associated class label of a tuple of which class label is unknown. In the first level, we design the classification function to specify the data classes. Typically, this mapping or functions are represented in the form of classification rules, decision trees, or mathematical formula etc. These rules are used to categorize the future data tuples, as to well as provide deeper insight into the data contents.

Second step classification model is used for classification of data. First, the predictive accuracy of the classifier is estimated to check the correctness of the classifier. If we use the training set to measure the accuracy of the classifier's, this is likely be come up with 100% accuracy, because the classifier is designed from the same dataset.Therefore, in some datasets, apart from the training dataset one more dataset is provided, known as testing data. They are independent of the training datasets, meaning that they were not used in the construction of the classifier. By using this dataset the classification model is tested for its accuracy and correctness.

The accuracy of a classifier on a given test set is the ratio of number of data that are correctly classified and the total number of data in the test set. For computation of correctness, the associated class label of each test tuple is compared with the computed class label obtained by the classifier. If the accuracy of the classifier is acceptable, then the classifier can be used for future classification of data.



CHECK YOUR PROGRESS

Q.1: Classification is .....

a) Supervised learning b) Unsupervised learning

c) Semi supervised learning d) None of the above

Q.2: Classification classifies the data and assigns the ..... of the data.

a) Data type b) File type

c) Class label d) Classification algorithm

Q.3: Classification and regression are used synonymously.

a) True b) False

Q.4: Testing dataset is used to design the classifier

a) True b) False

Q.5: Data classification is a two-step process, ... and a classification step.

a) Testing Phase b) Training Phase

c) Learning Phase d) None of the above

10.4 DECISION TREE

Decision Tree Analysis is a general, predictive modelling tool that has applications spanning a number of different areas. In general, decision trees are constructed via an algorithmic approach that identifies ways to split a data set based on different conditions. It is one of the most widely used and practical methods for supervised learning. Decision Trees are a non-parametric supervised learning method used for both classification and regression tasks. The goal is to create a model that predicts the value of a

target variable by learning simple decision rules inferred from the data features.

A decision tree is a tree-like graph with nodes representing the place where we pick an attribute and ask a question; edges represent the answers to the question; and the leaves represent the actual output or class label. They are used in non-linear decision making with simple linear decision surface.

Decision trees classify the problem dataset by sorting them down the tree from the root to some leaf node, with the leaf node providing the classification to the problem dataset. Each node in the tree acts as a test case for some attribute, and each edge descending from that node corresponds to one of the possible answers to the test case. This process is recursive in nature and is repeated for every subtree.

Let us discuss the decision tree problem with the following example. Let's assume we want to play badminton on a particular day — how we will decide whether to play or not. Let's say we go out and check if it's hot or cold, check the speed of the wind and the humidity, how the weather is i.e. is it sunny, cloudy or rainy. We take all these factors into account to decide whether we will play or not.

Let us assume that we record all these data for last 10 days as in the table 10.1 given below.

Table 10.1 Observations of the last ten days

Day	Weather	Temperature	Humidity	Wind	Play?
1	Sunny	Hot	High	Weak	No
2	Cloudy	Hot	High	Weak	Yes
3	Sunny	Mild	Normal	Strong	Yes
4	Cloudy	Mild	High	Strong	Yes
5	Rainy	Mild	High	Strong	No
6	Rainy	Cool	Normal	Strong	No
7	Rainy	Mild	High	Weak	Yes
8	Sunny	Hot	High	Strong	No
9	Cloudy	Hot	Normal	Weak	Yes
10	Rainy	Mild	High	Strong	No

Now, we may use this table to decide whether to play or not. But, if the weather pattern on that particular day does not match with any of rows in the table then it creates a problem. A decision tree would be a great way to represent data like this because it takes into account all the possible paths that can lead to the final decision by following a tree-like structure.

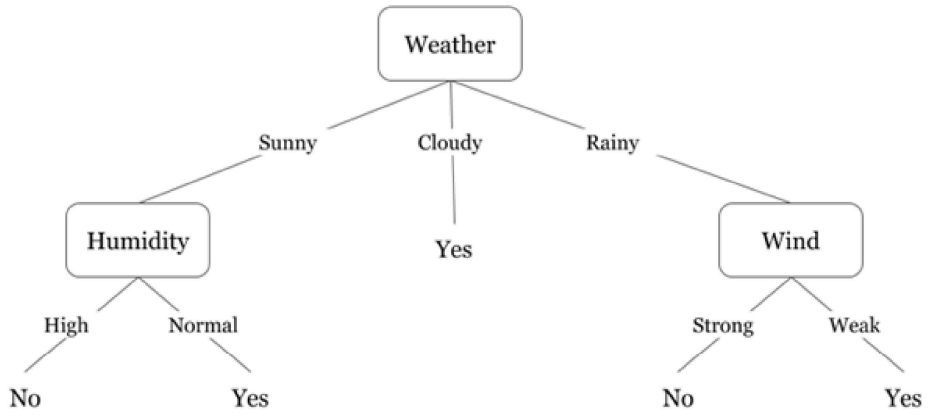


Figure 10.1: A decision tree for the concept play badminton

Figure 10.1 illustrates a learned decision tree. We can see that each node represents an attribute or feature and the branch from each node represents the outcome of that node. In each situation the leaves of the tree represents the final decision. If features are continuous; internal nodes can test the value of a feature against a threshold (see Figure. 10.2)

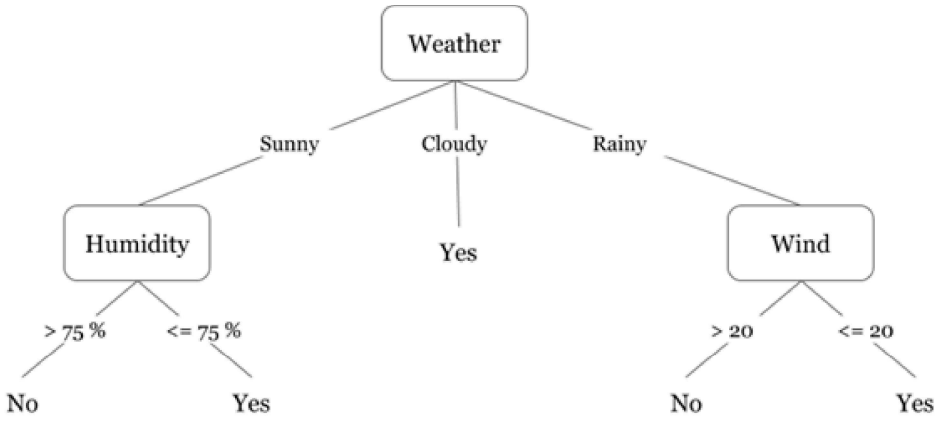


Figure 10.2: Decision tree for Badminton Play (with continuous attributes)

Incorporating continuous valued attributes: Our initial example is restricted to attributes that take on a discrete set of values. One way to make the decision tree more useful is with continuous variables. Let's take

our example of *Play Badminton* the temperature is continuous; we could test the information gain of certain partitions of the temperature values, such as temperature > 42.5. Typically, whenever the classification changes from no to yes or yes to no, the average of the two temperatures is taken as a potential partition boundary.

Table 10.2: Observations of the last ten days

Day	Weather	Temperature	Humidity	Wind	Play?
1	Sunny	80	High	Weak	No
2	Cloudy	66	High	Weak	Yes
3	Sunny	43	Normal	Strong	Yes
4	Cloudy	82	High	Strong	Yes
5	Rainy	65	High	Strong	No
6	Rainy	42	Normal	Strong	No
7	Rainy	70	High	Weak	Yes
8	Sunny	81	High	Strong	No
9	Cloudy	69	Normal	Weak	Yes

In the above data since 42 corresponds to No and 43 corresponds to Yes, 42.5 becomes a candidate. If any of the partitions end up exhibiting the greatest information gain, then it is used as an attribute and temperature is removed from the set of potential attributes to split on.

10.4.1 Decision Tree Algorithms

There are multiple algorithms written to build a decision tree, which can be used according to the problem characteristics we are trying to solve. A few of the commonly used algorithms are listed below:

- ID3
- C4.5
- CART
- CHAID (CHi-squared Automatic Interaction Detector)
- MARS
- Conditional Inference Trees

Though the methods are different for different decision tree building algorithms but all of them work on the principle of greediness.

Algorithms try to search for a variable which gives the maximum information gain or divides the data in the most homogenous way.

The basic algorithm used in decision trees is known as the ID3 (by Quinlan) algorithm. The ID3 algorithm builds decision trees using a top-down, greedy approach. Briefly, the steps to the algorithm are:

- Select the best attribute $\rightarrow$ A
- Assign A as the decision attribute (test case) for the **NODE**.
- For each value of A, create a new descendant of the **NODE**.
- Sort the training examples to the appropriate descendant node leaf.
- If examples are perfectly classified, then STOP else iterate over the new leaf nodes.

10.5 ATTRIBUTE SELECTION MEASURE

An attribute selection measure is a heuristic for selecting the splitting criterion that "best" separates a given data partition, D, of class-labelled training tuples into individual classes. If we were to split D into smaller partitions according to the outcomes of the splitting criterion, ideally each partition would be pure (i.e., all the tuples that fall into a given partition would belong to the same class). Conceptually, the "best" splitting criterion is the one that most closely results in such a scenario. Attribute selection measures are also known as splitting rules because they determine how the tuples at a given node are to be split.

The attribute selection measure provides a ranking for each attribute describing the given training tuples. The attribute having the best score for the measure is chosen as the splitting attribute for the given tuples. If the splitting attribute is continuous-valued or if we are restricted to binary trees, then, respectively, either a split point or a splitting subset must also be determined as part of the splitting criterion. The tree node created for partition D is labelled with the splitting criterion; branches are grown for each outcome of the criterion, and the tuples are partitioned accordingly. This section describes three popular attribute selection measures– *information gain*, *gain ratio*, and *Gini index*.

Now, the next big question is how to choose the best attribute. For ID3, the best attribute is selected based on the value of which attribute has the most *information gain*, a measure that expresses how well an attribute splits that data into groups based on classification.

10.5.1 Entropy & Information Gain

The word entropy is borrowed from Thermodynamics which is a measure of variability or chaos or randomness. Shannon extended the thermodynamic entropy concept in 1948 and introduced it into statistical studies and suggested the following formula for statistical entropy:

$$H = -\sum_{j=1}^{n} P_j \ln P_j$$

where, H is the entropy in the system which is a measure of randomness.

Assuming, rolling a fair coin and we want to know the entropy of the system. As per the formula given–

entropy would be equals to– *[0.5 ln(0.5) + 0.5 ln(0.5)]* = 0.69;

which is the maximum entropy which can occur in the system. In other words, there will be maximum randomness in our dataset if the probable outcomes have the same probability of occurrence.

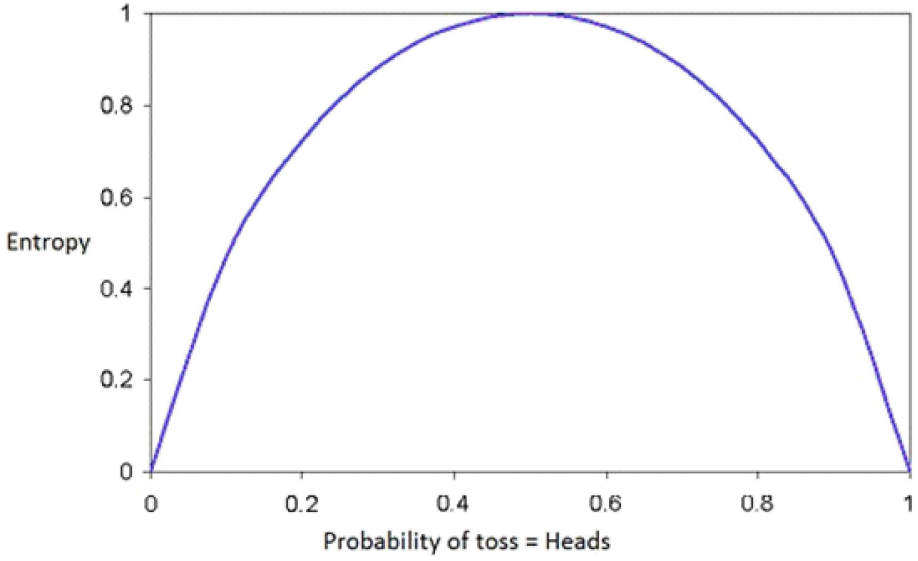


Figure 10.3: Example of Entropy Measure

The graph shown in figure 10.3 shows the variation of Entropy with the probability of a class. We can clearly see that Entropy is maximum when probability of either of the classes is equal. Now, we can understand that when a decision algorithm tries to split the data, it selects the variable which will give us maximum reduction in system Entropy.

Information Gain: The information gain is based on the decrease in entropy after a data-set is split on an attribute. Constructing a decision tree is all about finding attribute that returns the highest information gain.

$$Gain(T, X) = Entropy(T) – Entropy(T, X)$$

where *Entropy*(T, X) is the conditional entropy of T given the value of attribute X.

10.5.2 Alternative Measures for Selecting Attributes

The information gain formula used by ID3 algorithm treats all of the variables the same regardless of their distribution and their importance. This is a problem when it comes to continuous variables or discrete variables with many possible values because training examples may be few and far between for each possible value, which leads to low entropy and high information gain by virtue of splitting the data into small subsets but results in a decision tree that might not generalize well.

One way to avoid this is to use some other measure to find the best attribute instead of information gain. An alternative measure to information gain is *gain ratio*. Gain ratio tries to correct the information gain's bias towards attributes with many possible values by adding a denominator to *information gain* called *split information*. *Split Information* tries to measure how broadly and uniformly the attribute splits the data. One more measure for selecting the split attribute is Gini Index. Both measures are discussed below.

Gain Ratio: Soon after the development of entropy mathematicians realized that information gain is biased toward multi-valued attributes and to conquer this issue, "Gain Ratio" came into picture which is more reliable than information gain. The gain ratio can be defined as:

$$\text{Gain Ratio} = \frac{\text{Information Gain}}{\text{Split Info}}$$

where Split info can be defined as:

$$-\sum_{i=0}^{n} Di \log_2 Di$$

Assuming that we are dividing our variable into 'n' child nodes and Di represents the number of records going into various child nodes. Hence, gain ratio takes care of distribution bias while building a decision tree.

Gini Index: There is one more metric which can be used while building a decision tree i.e., Gini Index (Gini Index is mostly used in CART). Gini index measures the impurity of a data partition K, formula for Gini Index can be written down as:

$$\text{Gini}(K) = 1 - \sum_{i=1}^{n} P_i^2$$

where m is the number of classes, and P_i is the probability that an observation in K belongs to the class. Gini Index assumes a binary split for each of the attribute in S, let say T_1 & T_2. The Gini index of K given this partitioning is given by:

$$\text{Gini}_S(K) = \frac{T_1}{T}\text{Gini}(T_1) + \frac{T_2}{T}\text{Gini}(T_2)$$

which is nothing but a weighted sum of each of the impurities in split nodes. The reduction in impurity is given by:

$$\text{Gini}(K) - \text{Gini}_S(K)$$

Similar to Information Gain & Gain Ratio, split which gives us maximum reduction in impurity is considered for dividing our data.

Here, we have discussed all 3 commonly used metrics. However, confusion arises when we have to choose any one of them. There are a few drawbacks associated with all 3 of the metrics. These are summarized in the table below:

Table 10.3: Metrics and their drawbacks

Metrics	Drawback
Information Gain	Information Gain is biased towards multivariate attributes.
Gain Ratio	Gain Ratio generally prefers the unbalanced split of data where one of the child node has more number of entries compared to the others.
Gini Index	With more than 2 categories in the dataset, Gini Index gives unfavorable results. Apart from that it favors the split which results into equal sized children.

Many other attribute selection measures have been proposed. CHAID, a decision tree algorithm that is popular in marketing, uses an attribute selection measure that is based on the statistical 2 test for independence. Other measures include C-SEP (which per-forms better than information gain and the Gini index in certain cases) and G-statistic (an information theoretic measure that is a close approximation to 2 distribution).

Attribute selection measures based on the Minimum Description Length (MDL) principle have the least bias toward multivalued attributes. MDL-based measures use encoding techniques to define the "best" decision tree as the one that requires the fewest number of bits to both (1) encode the tree and (2) encode the exceptions to the tree(i.e., cases that are not correctly classified by the tree). Its main idea is that the simplest of solutions is preferred.

Other attribute selection measures consider multivariate splits (i.e., where the partitioning of tuples is based on a combination of attributes, rather than on a single attribute). The CART system, for example, can find multivariate splits based on a linear combination of attributes. Multivariate splits are a form of attribute (or feature) construction, where new attributes are created based on the existing ones.

"Which attribute selection measure is the best?" All measures have some bias. It has been shown that the time complexity of decision

tree induction generally increases exponentially with tree height. Hence, measures that tend to produce shallower trees may be preferred. However, some studies have found that shallow trees tend to have a large number of leaves and higher error rates. Despite several comparative studies, no one attribute selection measure has been found to be significantly superior to others. Most measures give quite good results.

10.6 DRAWBACKS OF USING DECISION TREES

Till now we have talked about various benefits of Decision Trees. But there are a few drawbacks or precautions which we should be aware of before going ahead with Decision trees. These are:

- Decision trees are susceptible to change in data; Even a small change in data can result into a completely new tree structure
- Decision trees tend to overfit but this can be overcome by pruning the trees
- We might face a problem when we are trying to do an out of sample testing or prediction.



CHECK YOUR PROGRESS

Q.6: A ..... is a decision support tool that uses a tree-like graph or model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility.

a) Decision tree b) Graphs

c) Trees d) Neural Networks

Q.7: Which data mining technique is most suitable for classification?

a) Clustering b) Association rule mining

c) Decision Tree d) None of the above

Q.8: ID3 is a .....

a) Clustering algorithm

b) Association rule mining algorithm

c) Decision Tree algorithm

d) Functional Algorithm

Q.9: ID3 uses .....

a) Gini Index b) Information Gain

c) Gain Ratio d) G-statistic

Q.10: Which of the following is not a decision tree building algorithm?

a) CART b) CHAID c) C4.5 d) K-Means



10.7 LET US SUM UP

- Classification is a data analysis technique used to classify large amount of data into different classes.
- This technique is used to predict category of data and assign the class labels.
- Classification has numerous applications, including fraud detection, target marketing, performance prediction, manufacturing, and medical diagnosis etc.
- Regression analysis is a statistical methodology that is most often used for numeric prediction; hence, classification and regression are used synonymously.
- Data classification is a two-step process, learning step and a classification step.
- In the learning step the classification model is constructed and in the classification step predicts class labels for given data.
- In the first step, a classifier is built using a predetermined set of data classes or concepts also known as training step.
- Since the class label of each training tuple is provided in the training dataset so this step is also known as supervised learning.

- In the second step the mappings of first step are used to predict the associated class label of a new tuple of which class label is unknown.
- The accuracy of a classifier on a given test set is the ratio of number of data that are correctly classified and the total number of data in the test set.
- A decision tree is a tree-like graph with nodes representing the place where we pick an attribute and ask a question; edges represent the answers to the question; and the leaves represent the actual output or class label.
- Decision trees classify the problem dataset by sorting them down the tree from the root to some leaf node, with the leaf node provide the class of the input data.
- There are multiple algorithms available for building a decision tree- ID3, C4.5, CART, CHAID, MARS etc.
- An attribute selection measure is a heuristic for selecting the splitting criterion of the decision tree.
- Three popular attribute selection measures are– *information gain*, *gain ratio*, and *Gini index*.
- Many other attribute selection measures also present like CHAID, C-SEP, G-statistic, MDL etc.
- Disadvantage of Decision Tree includes:
  - Decision trees are susceptible to change in data; Even a small change in data can result into a completely new tree structure
  - Decision trees tend to overfit but this can be overcome by pruning the trees
  - We might face a problem when we are trying to do an out of sample testing or prediction.



10.8 FURTHER READING

1) Aggarwal, C. C. (2015). *Data Mining: The Textbook*. Springer.
2) Han, J., Pei, J. & Kamber, M. (2011). *Data Mining: Concepts and Techniques*. Elsevier.

3) Tan, P. N. (2018). *Introduction to Data Mining*. Pearson Education India.



10.9 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q. No. 1: (a) **Ans. to Q. No. 2:** (c)

Ans. to Q. No. 3: (a) **Ans. to Q. No. 4:** (b)

Ans. to Q. No. 5: (c) **Ans. to Q. No. 6:** (a)

Ans. to Q. No. 7: (c) **Ans. to Q. No. 8:** (c)

Ans. to Q. No. 9: (b) **Ans. to Q. No. 10:** (d)



10.10 MODEL QUESTIONS

Q.1: What is Classification? Explain briefly.

Q.2: What is Decision Tree? List all the different decision tree building algorithms.

Q.3: What is information selection measure? Explain gini index?

Q.4: What is entropy and information gain? What is the use of it? Explain briefly.

Q.5: What are the disadvantages of using decision tree? Explain.

*** ***** ***

UNIT 11: PREDICTIONS

UNIT STRUCTURE

11.1 Learning Objectives

11.2 Introduction

11.3 Prediction Techniques

11.4 Statistical (Bayesian) Classification

11.5 Bayesian Networks

11.6 Instance-Based Methods (Nearest Neighbor)

11.6.1 K-Nearest Neighbor Method

11.7 Let Us Sum Up

11.8 Further Reading

11.9 Answers to Check Your Progress

11.10 Model Questions

11.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- define prediction
- describe prediction methods
- define Bayes' Theorem
- describe Bayesian Classification and Bayesian naïve classification
- describe Bayesian Belief Networks
- describe instance based method like k-nearest neighbor method.

11.2 INTRODUCTION

In the previous unit, we have discussed topics like classification, decision tree and attribute selection measures. In this unit, we go through the definition of prediction, different methods for prediction and where we use prediction. We will go through the Bayes' Theorem, Bayesian Classification, Bayesian naïve classification, Bayesian Belief Networks and what mechanism is in it. We will also discuss Lazy learner method specially k-nearest neighbor method for classification. In the next unit, we will explore the concept of classification in detail.

Prediction in data mining is to identify data points purely on the description of another related data value. We get the different prediction techniques like neural network, Bayesian classifier, decision tree, nearest neighbor, support vector machine, multiple linear regression etc. Bayesian classification is a statistical classifier. They can predict class membership probabilities. Bayesian classification based on Baye's theorem which gives the two types of probability: Posterior Probability [P(H|X)] and Prior Probability [P(H)].

Bayesian belief networks is probabilistic graphical models, which allow the representation of dependencies among subsets of attributes. Bayesian belief networks can be used for classification from which we get directed acyclic graph and conditional probability table.

The *k*-nearest neighbors algorithm (*k*-NN) is a non-parametric method used for classification and regression. In both cases, the input consists of the *k* closest training examples in the feature space. The output depends on whether *k*-NN is used for classification.

11.3 PREDICTION TECHNIQUES

Prediction in data mining is to identify data points purely on the description of another related data value. It is not necessarily related to future events but the used variables are unknown. Prediction derives the relationship between a thing you know and a thing you need to predict for future reference.

With the help of the following example we can explain Prediction. Suppose the marketing manager needs to predict how much a given customer will spend during a sale at his company. In this example, we are bothered to predict a numeric value, so that upcoming sale amount can be planned accordingly. Therefore, the data analysis task is an example of numeric prediction. In this case, a model or a predictor will be constructed that predicts a continuous-valued-function or an ordered value.

Prediction Techniques:

1) **Neural Network:** Artificial neural networks are based on the operation and structure of the human brain. A main feature of neural

networks is an iterative learning process in which data samples are presented to the network one at a time, and the weights are adjusted to predict the correct class label. Advantages of neural networks include their high tolerance to noisy data as well as their ability to classify patterns on which they have not been trained.

2) **Bayesian Classifier:** It is a statistical classification approach based on the Bayes theorem to calculate probability of A given B, P (B given A) = P(A and B)/P (A) the algorithm counts the number of cases where A and B occurs simultaneously and divides it by the number of cases where A alone occurs. Let X be a data tuple, X is considered "Evidence", in Bayesian terms. Let H be some hypothesis, such that the data tuple X belongs to class C. P (H|X) is posterior probability, of H conditioned on X. P (H) is the prior probability of H in contract.
3) **Decision Tree:** Decision tree uses the simple divide-and conquer algorithm. In these tree structures, leaves represent classes and branches signify conjunctions of features that lead to those classes. The attribute that most effectively splits samples into different classes is chosen, at each node of the tree. A path to a leaf from the root is found depending on the assessment of the predicate at each node that is visited, to predict the class label of an input. Decision tree is a fast and easy method.
4) **Support Vector Machine:** Normally SVM is the classification technique. Initially it is developed for binary type classification but later extended to multiple classifications. This SVM creates the hyper plane on the original inputs for effective separation of data points.
5) **Multiple Linear Regression:** This method is performed on a dataset to predict the response variable based on a predictor variable or is used to study the relationship between a response and predictor variable, for example, student test scores compared to demographic information such as income, education of parents, etc.
6) **k-Nearest Neighbors:** Like the classification method with the same name above, this prediction method divides a training dataset into groups of k observations using a Euclidean Distance measure

to determine similarity between "neighbors". These groups are used to predict the value of the response for each member of the validation set.

11.4 BAYESIAN (STATISTICAL) CLASSIFICATION

Bayesian classification is a statistical classifier. They can predict class membership probabilities. Bayesian classification is based on Bayes' theorem. Bayes' Theorem is named after Thomas Bayes, a nonconformist English clergyman who worked on probability and decision theory in the 18th century. There are two types of probabilities–

- Posterior Probability [P(H|X)]
- Prior Probability [P(H)]

where X as a data tuple belongs to the specified class C and H is some hypothesis.

According to Bayes' Theorem, P(H|X)= P(X|H)P(H) / P(X)

Naïve Bayesian Classification: There is not a single algorithm for training such classifiers, but a family of algorithms based on a common principle "all naive Bayes classifiers assume that the value of a particular feature is independent of the value of any other feature, given the class variable". The naïve classifier or simple classifier works as follows:

1) Let D be a training set of tuples and their associated class labels, each tuple is represented by an n-dimensional attribute vector, X=(x_1, x_2, …, …, ..., x_n), depicting measurements made on the tuple from n attributes respectively A_1, A_2, ..., ..., ..., A_n.
2) Suppose there are m classes C_1, C_2, …, …, ..., C_m. given a tuple X the classifier will predict that X belongs to the class having the highest posterior probability, conditioned on X. The naïve Bayesian classifier predict that tuple X belongs to the class C_i if and only if,

   $P(C_i|X) > P(C_j|X)$ for $1d \leq jd \leq m, j \neq i$

   Thus, we maximize $P(C_i|X)$. the class C_i for which $P(C_i|X)$ is maximized is called the maximum posteriori hypothesis. By Bayes' theorem, $P(C_i|X) = P(X|C_i)P(C_i) / P(X)$

3) As P(X) is constant for all classes, only $P(X|C_i)P(C_i)$ needs to be maximized. If the class prior probabilities are not known, then commonly assumed that all classes are equals. So we maximized $P(X|C_i)$, otherwise maximize $P(X|C_i)P(C_i)$.

4) If the dataset consists of many attributes, then it is extremely expensive to compute $P(X|C_i)$. To reduce computation, assume class conditional independence which explain that attributes values are conditionally independent of one another.

$$P(X|C_i)\,P(C_i) = \prod_{k=1}^{n} P(x_k|C_i)\,P(C_i)$$

$$= P(x_1|C_i)x\ P(x_2|C_i)x\ \ldots,\ \ldots,\ \ldots,\ x\ P(x_n|C_i)$$

Here x_k refers to the value of attribute A_k for tuple X. To compute $P(X|C_i)$ we consider the following:

a) If A_k is categorical, then $P(x_k|C_i)$ is the number of tuples of class C_i in D having the value x_k for A_k, divided by $|C_{i,D}|$, the number of tuples of class C_i in D.

b) If A_k is continuous value, then, $P(x_k|C_i)=g(x_k,\mu C_i, \sigma C_i)$

Considering that continuous value attribute have a Gaussian distribution with a mean μ and standard deviation σ, defined by,

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma}} e^{-(x-\mu)^2/2\sigma^2}$$

5) To predict the class label of X, $P(X|C_i)P(C_i)$ is evaluated for each class C_i. The classifier predicts that the class label of tuple X is the class C_i if and only if,

$P(X|C_i)P(C_i) > P(X|C_j)P(C_j)$ for $1d \le jd \le m, j \ne i$



CHECK YOUR PROGRESS

Q.1: What is the use of prediction?

.....

.....

Q.2: Name the different techniques of prediction.

.....

.....

.....

Q.3: What is the main feature of neural networks?

.....

.....

.....

Q.4: What is the advantage of neural network?

.....

.....

.....

Q.5: What is support vector machine?

.....

.....

11.5 BAYESIAN NETWORKS

Bayesian belief networks is probabilistic graphical models, which unlike naïve Bayesian classifiers allow the representation of dependencies among subsets of attributes. Bayesian belief networks can be used for classification.

Concepts and Mechanisms: The naïve Bayesian classifier makes the assumption of class conditional independence, that is, given the class label of a tuple, the values of the attributes are assumed to be conditionally independent of one another. This simplifies computation. When the assumption holds true, then the naïve Bayesian classifier is the most accurate in comparison with all other classifiers. In practice, however, dependencies can exist between variables. Bayesian belief networks specify joint conditional probability distributions. They allow class conditional independencies to be defined between subsets of variables. They provide a graphical model of causal relationships, on which learning can be performed. Trained Bayesian belief networks can be used for classification. Bayesian belief networks are also known as belief networks, Bayesian networks, and probabilistic

networks. For brevity, we will refer to them as belief networks. A belief network is defined by two components–

1) a directed acyclic graph and
2) a set of conditional probability tables

Directed Acyclic Graph: Each node in the directed acyclic graph represents a random variable. The variables may be discrete or continuous-valued. They may correspond to actual attributes given in the data or to "hidden variables" believed to form a relationship (e.g., in the case of medical data, a hidden variable may indicate a syndrome, representing a number of symptoms that, together, characterize a specific disease). Each arc represents a probabilistic dependence. If an arc is drawn from a node Y to a node Z, then Y is a parent or immediate predecessor of Z, and Z is a descendant of Y. Each variable is conditionally independent of its non-descendants in the graph, given its parents.

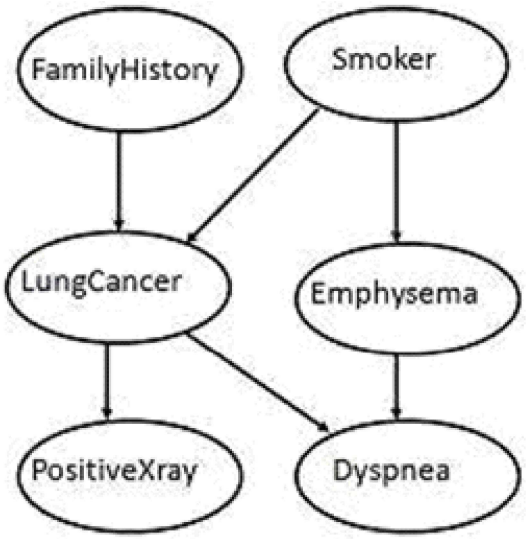


Figure 11.1: Example of Directed Acyclic Graph

For example, having lung cancer is influenced by a person's family history of lung cancer and whether the person is smoker or not. Variable PositiveXRay is independent of whether the patient has a family history of lung cancer or is a smoker, given that we know the patient has lung cancer. In other words, once we know the outcome of the variable LungCancer, then the variables FamilyHistory and Smoker do not provide any additional information regarding PositiveXRay. The arcs also show that the variable

LungCancer is conditionally independent of Emphysema, given its parents, FamilyHistory and Smoker.

Conditional Probability Table: The conditional probability table for the values of the variable LungCancer (LC) showing each possible combination of the values of its parent nodes, FamilyHistory (FH), and Smoker (S) is as follows:

	FH,S	FH,-S	-FH,S	-FH,S
LC	0.8	0.5	0.7	0.1
-LC	0.2	0.5	0.3	0.9

A belief network has one conditional probability table (CPT) for each variable. The CPT for a variable Y specifies the conditional distribution P(Y|Parents(Y)), where Parents(Y) are the parents of Y. The conditional probability for each known value of LungCancer is given for each possible combination of the values of its parents. For instance, from the upper leftmost and bottom rightmost entries, respectively, we see that

P(LungCancer = yes|FamilyHistory = yes, Smoker = yes) = 0.8

P(LungCancer = no |FamilyHistory = no, Smoker = no) = 0.9.

Training Bayesian Belief Networks: In the learning or training of a belief network, a number of scenarios are possible. The network topology (or "layout" of nodes and arcs) may be constructed by human experts or inferred from the data. The network variables may be observable or hidden in all or some of the training tuples. The hidden data case is also referred to as missing values or incomplete data. Several algorithms exist for learning the network topology from the training data given observable variables.

Let D be a training set of data tuples, X1, X2, ..., ..., ..., X|D| . Training the belief network means that we must learn the values of the CPT entries. Let w_{ijk} be a CPT entry for the variable $Y_i = y_{ij}$ having the parents $U_i = u_{ik}$, where $w_{ijk} \equiv P(Y_i = y_{ij}|U_i = u_{ik})$.

For example, if w_{ijk} is the upper leftmost CPT entry of above Conditional Probability Table, then Y_i is LungCancer; y_{ij} is its value, "yes"; U_i lists the parent nodes of Y_i, namely, {FamilyHistory, Smoker}; and u_{ik} lists the values of the parent nodes, namely, {"yes", "yes"}. The w_{ijk} are viewed

as weights. The weights are initialized to random probability values. A gradient descent strategy performs greedy hill-climbing. At each iteration, the weights are updated and will eventually converge to a local optimum solution.

For our problem, we maximize $P_w(D)$. This can be done by following the gradient of $\ln P_w(S)$, which makes the problem simpler. Given the network topology and initialized w_{ijk}, the algorithm proceeds as follows:

1) **Compute the Gradients:** For each i, j, k, compute:

$$\frac{\partial \ln P_W(D)}{\partial W_{ijk}} = \sum_{d=1}^{|D|} \frac{P(Y_i = y_{ij}, U_i = u_{ik} \mid X_d)}{W_{ijk}} \qquad (11.1)$$

The probability on the right side of Eq. (11.1) is to be calculated for each training tuple, X_d, in D. For brevity, let's refer to this probability simply as p. When the variables represented by Y_i and U_i are hidden for some X_d, then the corresponding probability p can be computed from the observed variables of the tuple using standard algorithms for Bayesian network inference.

2) **Take a small step in the direction of the gradient:** The weights are updated by,

$$w_{ijk} \leftarrow w_{ijk} + (l)\partial \ln P_w(D)/\partial w_{ijk} \qquad (11.2)$$

where l is the learning rate representing the step size and $\partial \ln P_w(D)/\partial w_{ijk}$ is computed from Eq. (11.1). The learning rate is set to a small constant and helps with convergence.

3) **Renormalize the weights:** Because the weights w_{ijk} are probability values, they must be between 0.0 and 1.0, and P j w_{ijk} must equal 1 for all i, k.

11.6 INSTANCE-BASED METHODS (NEAREST NEIGHBOR)

In Lazy Learners method when given a set of training tuples, we construct a generalization (i.e., classification) model before receiving new tuples to classify. We can think of the learned model as being ready and eager to classify previously unseen tuples. Imagine a contrasting lazy approach, in which the learner instead waits until the last minute before doing

any model construction to classify a given test tuple. That is, when given a training tuple, a lazy learner simply stores it and waits until it is given a test tuple. Only when it sees the test tuple does it perform generalization to classify the tuple based on its similarity to the stored training tuples. Lazy learners do less work when a training tuple is presented and more work when making a classification or numeric prediction. Because lazy learners store the training tuples or "instances", they are also referred to as instance-based learners, even though all learning is essentially based on instances. When making a classification or numeric prediction, lazy learners can be computationally expensive. K-nearest neighbor method is example of lazy learner method.

11.6.1 K-nearest Neighbor Method

The k-nearest-neighbor method was first described in the early 1950s. The method is labor intensive when given large training sets, and did not gain popularity until the 1960s when increased computing power became available. It has since been widely used in the area of pattern recognition. Nearest-neighbor classifiers are based on learning by analogy, that is, by comparing a given test tuple with training tuples that are similar to it. The training tuples are described by n attributes. Each tuple represents a point in an n-dimensional space. In this way, all the training tuples are stored in an n-dimensional pattern space. When given an unknown tuple, a k-nearest-neighbor classifier searches the pattern space for the k training tuples that are closest to the unknown tuple. These k training tuples are the k "nearest neighbors" of the unknown tuple. "Closeness" is defined in terms of a distance metric, such as Euclidean distance. The Euclidean distance between two points or tuples, say, $X_1 = (x_{11}, x_{12}, ..., ..., ..., x_{1n})$ and $X_2 = (x_{21}, x_{22}, ..., ..., ..., x_{2n})$, is:

$$\text{dist}(X_1, X_2) = \sqrt{\sum_{i=1}^{n} (X_{1i} - X_{2i})^2} \qquad (11.3)$$

In other words, for each numeric attribute, we take the difference between the corresponding values of that attribute in tuple

X_1 and in tuple X_2, square this difference, and accumulate it. The square root is taken of the total accumulated distance count. Typically, we normalize the values of each attribute before using Eq. (11.3). This helps prevent attributes with initially large ranges (e.g., income) from outweighing attributes with initially smaller ranges (e.g., binary attributes). Min-max normalization, for example, can be used to transform a value v of a numeric attribute A to v' in the range [0, 1] by computing–

$$\bar{v} = \frac{v - \min_A}{\max_A - \min_A} \qquad (11.4)$$

where $\min_A$ and $\max_A$ are the minimum and maximum values of attribute A.

For k-nearest-neighbor classification, the unknown tuple is assigned the most common class among its k-nearest neighbors. When k = 1, the unknown tuple is assigned the class of the training tuple that is closest to it in pattern space. Nearest-neighbor classifiers can also be used for numeric prediction, that is, to return a real-valued prediction for a given unknown tuple. In this case, the classifier returns the average value of the real-valued labels associated with the k-nearest neighbors of the unknown tuple.

Distance calculation method for nominal attributes: It is a simple method used to compare the corresponding value of the attribute in tuple X_1 with that in tuple X_2. If the two are identical (e.g., tuples X_1 and X_2 both have the color blue), then the difference between the two is taken as 0. If the two are different, then the difference is considered to be 1.

Missing values: In general, if the value of a given attribute A is missing in tuple X_1 and/or in tuple X_2, we assume the maximum possible difference. Suppose that each of the attributes has been mapped to the range [0, 1]. For nominal attributes, we take the difference value to be 1 if either one or both of the corresponding values of A are missing. If A is numeric and missing from both tuples X_1 and X_2, then the difference is also taken to be 1. If only one value

is missing and the other is present and normalized, then we can take the difference to be either |1 – v' | or |0 – v' | (i.e., 1 – v' or v'), whichever is greater.

Calculation of number of neighbors(k): This can be determined experimentally. Starting with k = 1, we use a test set to estimate the error rate of the classifier. This process can be repeated each time by incrementing k to allow for one more neighbor. The k value that gives the minimum error rate may be selected. In general, the larger the number of training tuples means the larger the value of k. Nearest-neighbor classifiers use distance-based comparisons that intrinsically assign equal weight to each attribute. They therefore can suffer from poor accuracy when given noisy or irrelevant attributes. Nearest-neighbor classifiers can be extremely slow when classifying test tuples. If D is a training database of |D| tuples and k = 1, then O(|D|) comparisons are required to classify a given test tuple.

Other techniques to speed up classification time include the use of partial distance calculations and editing the stored tuples. In the partial distance method, we compute the distance based on a subset of the n attributes. If this distance exceeds a threshold, then further computation for the given stored tuple is halted, and the process moves on to the next stored tuple. The editing method removes training tuples that prove useless. This method is also referred to as pruning or condensing because it reduces the total number of tuples stored.



CHECK YOUR PROGRESS

Q.6: Define Multiple Linear Regression method.

.....

.....

.....

Q.7: Define Bayesian network.

.....

.....

Q.8: What are the components of Bayesian belief networks?

.....

.....

Q.9: What are the steps of gradient descent strategy?

.....

.....

.....

Q.10: What is the use of Lazy Learner method?

.....

.....

.....



11.7 LET US SUM UP

- Neural network, Bayesian classification, decision tree, nearest neighbor, support vector machine, multiple linear regressions are different prediction methods.
- Bayes' theorem gives posterior and prior probability.
- Naïve Bayesian classification is based on 'the value of particular feature is independent of the value of any other feature given the class value'.
- Decision tree is based on divide and conquer method.
- Bayesian Network is a probabilistic graphical model.
- Directed Acyclic Graph and Conditional Probability Table can achieved fron Bayesian Network.
- DAG represents probability dependency.
- CTP values show each possible combination.
- Calculation of k in k-nearest neighbor classifier is starts from 1 and process will be repeat with increment of k value.



11.8 FURTHER READING

1) Han, J., Pei, J. & Kamber, M. (2011). *Data Mining: Concepts and Techniques*. Elsevier.

2) Pujari, A. K. (2001). *Data Mining Techniques*. Universities Press.



11.9 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q. No. 1: Prediction in data mining is to identify data points purely on the description of another related data value.

Ans. to Q. No. 2: Different prediction techniques are neural network, Bayesian classifier, decision tree, nearest neighbor, support vector machine, multiple linear regressions.

Ans. to Q. No. 3: Main feature of neural networks is an iterative learning process in which data samples are presented to the network one at a time, and the weights are adjusted to predict the correct class label.

Ans. to Q. No. 4: Advantages of neural networks include their high tolerance to noisy data, and ability to classify patterns on which they have not been trained.

Ans. to Q. No. 5: Support Vector Machine technique is developed for binary type classification later extended to multiple classifications.

Ans. to Q. No. 6: Multiple Linear Regression method is performed on a dataset to predict the response variable based on a predictor variable or used to study the relationship between a response and predictor variable.

Ans. to Q. No. 7: Bayesian belief networks is probabilistic graphical models, which allow the representation of dependencies among subsets of attributes.

Ans. to Q. No. 8: A belief network is defined by two components– a directed acyclic graph and a set of conditional probability tables.

Ans. to Q. No. 9: Gradient descent strategy are consists of three steps: calculate the gradient, take a small step in the direction of the gradient and renormalized the weight.

Ans. to Q. No. 10: Lazy Learners when given a set of training tuples, will construct a generalization model before receiving new uples to classify.



11.9 MODEL QUESTIONS

Q.1: What is prediction?

Q.2: Write about the different techniques of prediction.

Q.3: Explain the Bayes' Theorem.

Q.4: What is the Naïve Bayesian Classification?

Q.5: What is mechanism used in Bayesian Networks?

Q.6: Explain directed acyclic graph with example.

Q.7: What is conditional probability table? Explain with example.

Q.8: What is training process related to Bayesian Networks?

Q.9: Explain the k-nearest neighbor classification method for prediction.

*** ***** ***

UNIT 12: EVALUATION

UNIT STRUCTURE

12.1 Learning Objectives

12.2 Introduction

12.3 Training and Testing

12.4 Evaluating Accuracy of a Classifier

12.5 Cross-Validation

12.6 Combining Multiple Models

12.7 Let Us Sum Up

12.8 Further Reading

12.9 Answer to Check Your Progress

12.10 Model Questions

12.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- define training data and testing data
- describe how to evaluate accuracy of a classifier
- define cross validation
- describe combination of multiple models.

12.2 INTRODUCTION

In the previous unit, we have learned about prediction, different methods for prediction and where we can use prediction. We have also learned in detail about Bayes' Theorem, Bayesian Classification, Bayesian naïve classification, Bayesian Belief Networks along with k-nearest neighbor method for classification. In this unit we will learn what is test data and training data and how they differ from each other. We will also learn how to calculate the accuracy and improve the accuracy of a classifier and how it is effective for future prediction. We will learn topics like cross validation and how we can perform statistical analysis for independent dataset. In the next unit, we will explore the concept of clustering.

When we have to build a classification model, some questions arise like what was the previous data and on the basis of what can we build the classification model. After building the classification model, we have to predict if it will be effective or not for the future business. On basis of that we predict whether the particular classifier is accurate or not. Classification Accuracy means the ratio of number of correct predictions to the total number of input samples. Accuracy can be measured with the help of confusion matrix, area under curve, F1 score etc.

Cross-validation is a technique that is used for the assessment of how the results of statistical analysis generalize to an independent data set. Cross-validation is largely used in settings where the target is prediction and it is necessary to estimate the accuracy of the performance of a predictive model. The prime reason for the use of cross-validation rather than conventional validation is that there is not enough data available for partitioning them into separate training and test sets.

To improve the accuracy of the classification, ensemble methods are introduced. The ensemble for classification is a composite model of classifier. The individual classifiers vote and a class label prediction is returned by the ensemble based on collection of votes. Ensemble tends to be more accurate than their component classifiers.

12.3 TRAINING AND TESTING

When we have to build a classification model, many questions are going through our mind. For example, suppose you used previous sales data to build a classifier to predict the purchase behavior of a customer. Classifier can predict the future customer purchasing behavior, that is, future customer data on which the classifier has not been trained. Even we can build more than one classifier using different methods and wish to compare their accuracy to predict the best classifier.

A training dataset is a dataset of examples used for learning that is to fit the parameters (e.g., weights) of a classifier. Most approaches that search through training data for empirical relationship stend to overfit the data, meaning that they can identify and exploit apparent relationships in the training data that do not hold in general.

A test dataset is a dataset that is independent of the training dataset, but that follows the same probability distribution as the training dataset. A model that fits to the training dataset also fits the test dataset well, minimal over-fitting has taken place. A better fitting of the training dataset as opposed to the test dataset usually points to over-fitting. A test set is therefore a set of examples used only to assess the performance of a fully specified classifier.

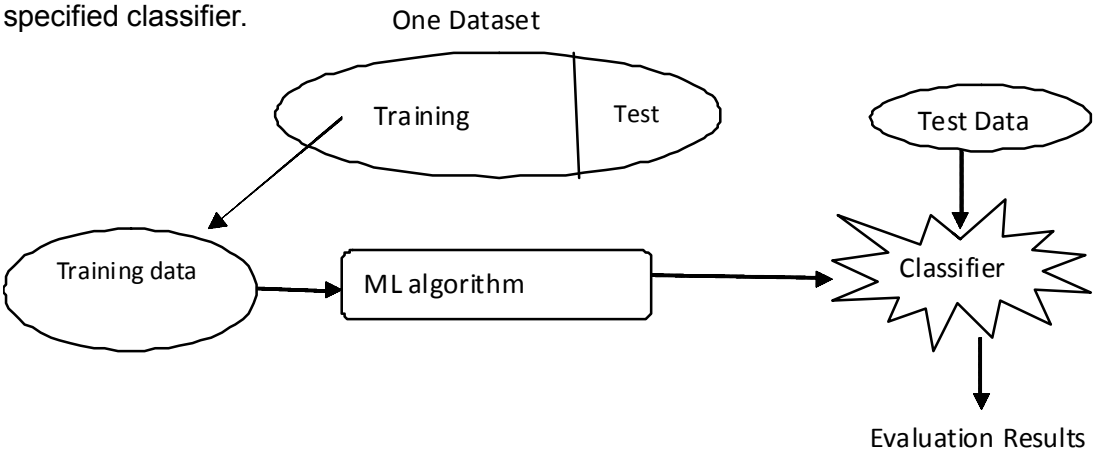


Figure 12.1: Example of Classifier

In data mining, the data is divided into training set (most of data) and testing set (smaller portion) after the model processed by using the training set. We test the model by making prediction against the test set based on the value that determined for training set. If your division sets are n numbers this process is repeated for n times; for example if your data divided into four training set and one testing set, the process repeated for five times alternatively. Now it's easy to know whether your model guesses are correct or not.

12.4 EVALUATIING ACCURACY OF A CLASSIFIER

Classification accuracy is what we usually mean, when we use the term accuracy. It is the ratio of number of correct predictions to the total number of input samples.

$$\textbf{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions Made}}$$

It works well only if there are equal number of samples belonging to each class.

For example, consider that there are 98% samples of class A and 2% samples of class B in our training set. Then our model can easily get **98% training accuracy** by simply predicting every training sample belonging to class A.

When the same model is tested on a test set with 60% samples of class A and 40% samples of class B, then the **test accuracy would drop down to 60%**. Classification accuracy is great, but it gives us the false sense of achieving high accuracy.

The real problem arises, when the cost of misclassification of the minor class samples are very high. If we deal with a rare but fatal disease, the cost of failing to diagnose the disease of a sick person is much higher than the cost of sending a healthy person to more tests.

Confusion Matrix: Confusion Matrix as the name suggests gives us a matrix as output and describes the complete performance of the model.

Let us assume we have a binary classification problem. We have some samples belonging to two classes: YES or NO. Also, we have our own classifier which predicts a class for a given input sample. On testing our model on 165 samples, we get the following result.

n = 165	**Predicted: NO**	**Predicted: YES**
Actual: NO	**50**	**10**
Actual: YES	**5**	**100**

Figure 12.2: Confusion Matrix

There are 4 important terms:

- **True Positives:** The cases in which we predicted YES and the actual output was also YES.
- **True Negatives:** The cases in which we predicted NO and the actual output was NO.
- **False Positives:** The cases in which we predicted YES and the actual output was NO.

- **False Negatives:** The cases in which we predicted NO and the actual output was YES.

Accuracy for the matrix can be calculated by taking average of the values lying across the **"main diagonal"** i.e.,

$$\textbf{Accuracy} = \frac{\textbf{True Positives + False Negatives}}{\textbf{Total Number of Samples}}$$

$$\therefore \textbf{Accuracy} = \frac{\textbf{100 + 50}}{\textbf{165}} = \textbf{0.91}$$

Confusion Matrix forms the basis for the other types of metrics.

Area Under Curve: Area Under Curve(AUC) is one of the most widely used metrics for evaluation. It is used for binary classification problem. AUC of a classifier is equal to the probability that the classifier will rank a randomly chosen positive example higher than a randomly chosen negative example. Before defining AUC, let us understand two basic terms:

- **True Positive Rate (Sensitivity):** True Positive Rate is defined as *TP/ (FN*+TP). True Positive Rate corresponds to the proportion of positive data points that are correctly considered as positive, with respect to all positive data points.

$$\textbf{True Positive Rate} = \frac{\textbf{True Positive}}{\textbf{(False Negative + True Positive)}}$$

- **False Positive Rate (Specificity):** False Positive Rate is defined as *FP / (*FP+TN). False Positive Rate corresponds to the proportion of negative data points that are mistakenly considered as positive, with respect to all negative data points.

$$\textbf{False Positive Rate} = \frac{\textbf{False Positive}}{\textbf{(False Positive + True Negative)}}$$

False Positive Rate and True Positive Rate both have values in the range [0, 1]. FPR and TPR both are computed at threshold values such as (0.00, 0.02, 0.04, …, ..., ..., 1.00) and a graph is drawn. AUC is the area under the curve of plot False Positive Rate vs True Positive Rate at different points in [0, 1].

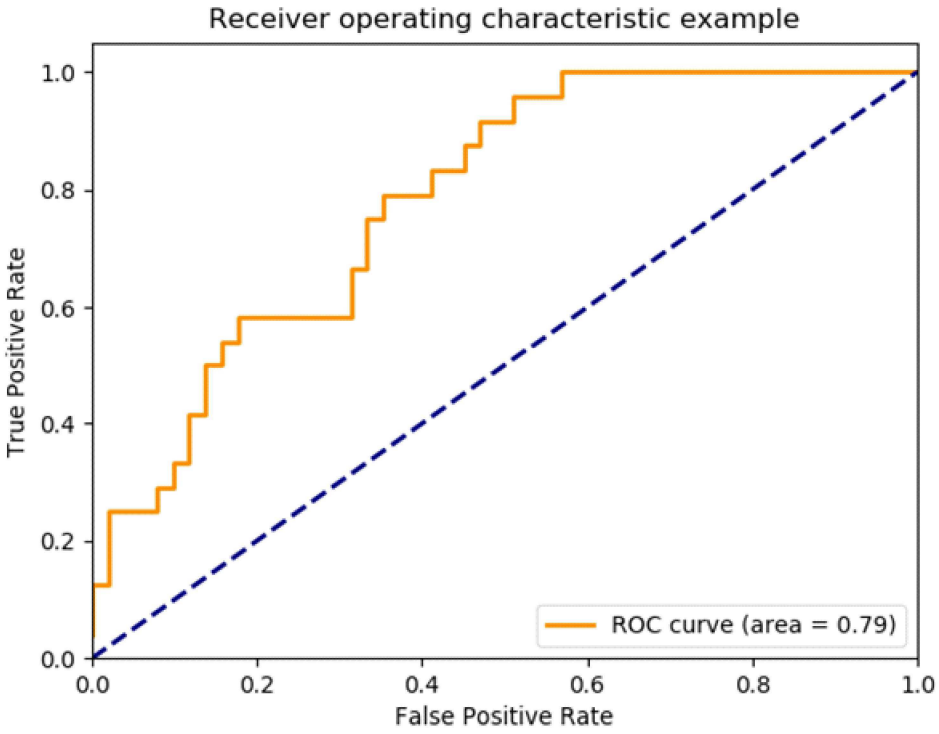


Figure 12.3: Graph of False Positive Rate versus True Positive Rate

As evident, AUC has a range of [0, 1]. The greater the value, the better is the performance of our model.

F1 Score: *F1 Score is used to measure a test's accuracy.* F1 Score is the Harmonic Mean between precision and recall. The range for F1 Score is [0, 1]. It tells you how precise your classifier is (how many instances it classifies correctly), as well as how robust it is (it does not miss a significant number of instances).

High precision but lower recall, gives you an extremely accurate, but it then misses a large number of instances that are difficult to classify. The greater the F1 Score, the better is the performance of our model. Mathematically, it can be expressed as:

$$F1 = 2 * \frac{1}{\frac{1}{precision} + \frac{1}{recall}}$$

F1 Score tries to find the balance between precision and recall.

- **Precision:** It is the number of correct positive results divided by the number of positive results predicted by the classifier.

$$\text{Precision} = \frac{\text{True Positive}}{\text{(False Negatives + True Positives)}}$$

- **Recall:** It is the number of correct positive results divided by the number of ***all*** relevant samples (all samples that should have been identified as positive).

$$\text{Recall} = \frac{\text{True Positive}}{\text{(True Positives + False Negatives)}}$$

Classifiers can be also compared with respect to the following aspects:

- **Speed:** This refers to the computational costs involved in generating and using the given classifier.
- **Robustness:** This is the ability of the classifier to make correct predictions given noisy data or data with missing values. Robustness is typically assessed with a series of synthesis data sets representing increasing degrees of noise of missing values.
- **Scalability:** This refers to the ability to construct the classifier efficiently given large amounts of data.
- **Interpretability:** This refers to the level of understanding and insight that is provided by the classifier or predictor. Interpretability is subjective and more difficult to assess.



CHECK YOUR PROGRESS

Q.1: What is test dataset?

.....

.....

.....

Q.2: What is precision?

.....

.....

.....

Q.3: What is confusion matrix?

.....

.....

Q.4: What is sensitivity and specificity?

.....

.....

.....

.....

.....

.....

.....

.....

Q.5: How can we compare the classifier?

.....

.....

12.5 CROSS VALIDATION

Cross-validation is a technique that is used for the assessment of how the results of statistical analysis generalize to an independent data set. Cross-validation is largely used in settings where the target is prediction and it is necessary to estimate the accuracy of the performance of a predictive model. The prime reason for the use of cross-validation rather than conventional validation is that there is not enough data available for partitioning them into separate training and test sets (as in conventional validation). This results in a loss of testing and modeling capability.

For a prediction problem, a model is generally provided with a data set of known data, called the training data set, and a set of unknown data against which the model is tested, known as the test data set. The target is to have a data set for testing the model in the training phase and then provide insight on how the specific model adapts to an independent data set. A round of cross-validation comprises the partitioning of data into complementary subsets, then performing analysis on one subset. After this, the analysis is validated on other subsets (testing sets). To reduce variability, many rounds of cross-validation are performed using many different partitions and then an average of the results is taken. Cross-validation is a powerful technique in the estimation of model performance technique.

K-fold Cross Validation: K-Fold Cross Validation is a common type of cross validation that is widely used in machine learning. K-fold cross validation is performed as per the following steps:

Step 1: Partition the original training data set into k equal subsets. Each subset is called a **fold**. Let the folds be named as f_1, f_2, …, ..., f_k.

Step 2: For i = 1 to i = k

- Keep the fold f_i as validation set and keep all the remaining *k–1* folds in the cross validation training set.
- Train your machine learning model using the cross validation training set and calculate the accuracy of your model by validating the predicted results against the validation set.

Step 3: Estimate the accuracy of your machine learning model by averaging the accuracies derived in all the *k* cases of cross validation.

In the k-fold cross validation method, all the entries in the original training data set are used for both training as well as validation. Also, each entry is used for validation just once.

Generally, the value of *k* is taken to be 10, but it is not a strict rule, and *k* can take any value.

Leave-one-out is special case k-fold cross validation where k is set to the number of initial tuples. That is only one sample is left out at the time of testing. In stratified cross validation, the folds are stratified so that class distribution of the tuples in each fold is approximately same as that of the initial data.

12.6 COMBAINING MULTIPLE MODELS

To improve the accuracy of the classification, ensemble methods are introduced. The ensemble for classification is a composite model of classifier. The individual classifiers vote and a class label prediction is returned by the ensemble based on collection of votes. Ensemble based results tend to be more accurate than their component classifiers.

Bootstrap: The bootstrap method is a re-sampling technique used to estimate statistics on a population by sampling a dataset with replacement.

T*he bootstrap is a widely applicable and extremely powerful statistical tool that can be used to quantify the uncertainty associated with a given estimator or statistical learning method.*

It can be used to estimate summary statistics such as the mean or standard deviation. It is used in applied machine learning to estimate the skill of machine learning models when making predictions on data not included in the training data.

Importantly, samples are constructed by drawing observations from a large data sample one at a time and returning them to the data sample after they have been chosen. This allows a given observation to be included in a given small sample more than once. This approach to sampling is called sampling with replacement.

The process for building one sample can be summarized as follows:

Step 1: Choose the size of the sample.

Step 2: While the size of the sample is less than the chosen size

- Randomly select an observation from the dataset
- Add it to the sample

Example 12.1: The bootstrap method can be used to estimate the quantity of a population. This is done by repeatedly taking small samples, calculating the statistics and taking the average of the calculated statistics. We can summarize this procedure as follows:

Step 1: Choose a number of bootstrap samples to perform

Step 2: Choose a sample size

Step 3: For each bootstrap sample

- Draw a sample with replacement with the chosen size
- Calculate the statistic on the sample

Step 4: Calculate the mean of the calculated sample statistics.

The procedure can also be used to estimate the skill of a machine learning model.

Boosting: Boosting is a machine learning ensemble meta-algorithm for primarily reducing bias, and also variance in supervised learning, and a family of machine learning algorithms that convert weak learners to strong ones. Boosting is based on the question posed by Kearns and Valiant (1988,

1989). "Can a set of **weak learners** create a single **strong learner**?" A weak learner is defined to be a classifier that is only slightly correlated with the true classification. In contrast, a strong learner is a classifier that is arbitrarily well-correlated with the true classification.

Bagging: Bootstrap Aggregation also known as bagging, is a powerful and simple ensemble method to improve the stability and accuracy of machine learning algorithms used in statistical classification and regression. Bagging is a special case of the model averaging approach.

An ensemble method is a technique that combines the predictions from many machine learning algorithms together to make more reliable and accurate predictions than any individual model. It means that we can say that prediction of bagging is very strong. The main purpose of using the bagging technique is to improve Classification Accuracy.

Description of the Bagging Technique: Given a standard training set of size n, bagging generates m new training sets, each of size n', by sampling from D uniformly and with replacement. By sampling with replacement, some observations may be repeated in each. If $n' = n$, then for large n the set is expected to have the fraction $(1 - 1/e)$ ($H \approx 63.2\%$) of the unique examples of D, the rest being duplicates. This kind of sample is known as a bootstrap sample. The m models are fitted using the above m bootstrap samples and combined by averaging the output (for regression) or voting (for classification).

For example, we have 1000 observations and 200 elements. In bagging, we will create several models with a subset of variables and a subset of observations. i.e., we might create 300 trees with 300 random variables and 20 observations in each tree. After that, we can average the results of all the 300 tree's (models) to get to our final prediction.



CHECK YOUR PROGRES

Q.6: What is the standard value for k in k fold cross validation method?

.....

.....

Q.7: What is fold?

.....

.....

Q.8: What is leave-one-out?

.....

.....

Q.9: What is bootstrap?

.....

.....

.....

.....



12.7 LET US SUM UP

- Dataset can be divided as test dataset and training dataset.
- A test dataset is a dataset that is independent of the training dataset, but that follows the same probability distribution as the training dataset.
- Confusion matrix describes the complete performance of model.
- Confusion Matrix is defined on the basis of true positive, true negative, false positive, false negative.
- True positive rate (Sensitivity) is defined as TP/(FN+TP).
- False positive rate (Specificity) is defined as FP/(FP+TN).
- F1 score measures the test's accuracy and it tries to find balance between recall and precision.
- Cross validation is used where target is prediction.

- To improve the accuracy of the classification, ensemble methods are introduced.



12.8 FURTHER READING

1) Han, J., Pei, J. & Kamber, M. (2011). *Data Mining: Concepts and Techniques*. Elsevier.
2) Pujari, A. K. (2001). *Data Mining Techniques*. Universities Press.



12.9 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q. No. 1: A test dataset is a dataset that is independent of the training dataset, but that follows the same probability distribution as the training dataset

Ans. to Q. No. 2: It is the number of correct positive results divided by the number of positive results predicted by the classifier.

Precision = True Positives/ (True Positives + False Positives)

Ans. to Q. No. 3: Confuse matrix gives a matrix as output and describes the complete performance of the model.

Ans. to Q. No. 4: True Positive Rate is defined as *TP/ (FN*+TP). True Positive Rate corresponds to the proportion of positive data points that are correctly considered as positive, with respect to all positive data points.

True Positive Rate = True Positive / (False Negative + True Positive)

False Positive Rate is defined as *FP / (*FP+TN). False Positive Rate corresponds to the proportion of negative data points that are mistakenly considered as positive, with respect to all negative data points.

False Positive Rate = False Positive / (False Positive + True Negative)

Ans. to Q. No. 5: Classifiers are compared with respect to the following aspects: Speed, robustness, scalability and interpretability.

Ans. to Q. No. 6: The value of *k* is taken to be 10 in K fold cross validation method, but it is not a strict rule, and *k* can take any value.

Ans. to Q. No. 7: Partition the original training data set into k equal subsets. Each subset is called a **fold.**

Ans. to Q. No. 8: Leave-one-out is special case k-fold cross validation where k is set to the number of initial tuples.

Ans. to Q. No. 9: The bootstrap method is a resampling technique used to estimate statistics on a population by sampling a dataset with replacement. The bootstrap is a widely applicable and extremely powerful statistical tool that can be used to quantify the uncertainty associated with a given estimator or statistical learning method.



12.10 MODEL QUESTIONS

Q.1: Differentiate between test dataset and training dataset.

Q.2: How can we calculate the accuracy of a classifier?

Q.3: What are the factors effected in accuracy of a classifier?

Q.4: Explain the k fold cross validation method.

Q.5: What is cross validation? How is cross validation effective for dataset?

Q.6: Explain the combined model of classifier.

Q.7: What is bootstrap?

Q.8: What is boosting and bagging?

Q.9: Differentiate between boosting and bagging.

*** ***** ***

UNIT 13: CLUSTERING

UNIT STRUCTURE

13.1 Learning Objectives

13.2 Introduction

13.3 Clustering

13.4 Classification of Various Clustering Algorithms

13.4.1 Partitioning Method

13.4.2 Hierarchical Method

13.5 Let Us Sum Up

13.6 Further Reading

13.7 Answers to Check Your Progress

13.8 Model Questions

13.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- define clustering
- describe various classification of clustering algorithm
- describe partitioning and hierarchical methods
- describe how K-means algorithm works
- describe DBSCAN algorithm.

13.2 INTRODUCTION

In the previous unit, we have learnt about training data and testing data. We have also learned about cross validation and different evaluation measures. In this unit, we will learn about different clustering techniques. We will also learn the various classification of clustering algorithms such as partition algorithm, hierarchical algorithm, divisive algorithm etc. We will also learn two different clustering algorithms K-means and DBSCAN algorithm in detail in this unit. In the next unit, we will explore the topic of web data and web mining.

13.3 CLUSTERING

Clustering is a usefull technique for the discovery of data distribution and pattern of given data. The main aim of clustering technique is to find both distributed and dense region in a data set. Clustering technique includes several classes of clustering algorithms. These algorithm attempts to automatically partition the data space into several region i.e to sparse or dense region. The main goal of clustering is to find all sets of similar examples in the data set. Following are the different objectives of clustering techniques:

- To uncover natural grouping.
- To initiate hypothesis about the data.
- To find consistent and valid organization of data.

Clustering has several applications. Following are some of them:

- Clustering analysis is broadly used in many applications such as market research, pattern recognition, data analysis, and image processing.
- Clustering can also help marketers to discover distinct groups in their customer base. And they can characterize their customer groups based on the purchasing patterns.
- Clustering also helps in the identification of areas of similar land use in an earth observation database. It also helps in the identification of groups of houses in a city according to house type, value, and geographic location.
- Clustering also helps in classifying documents on the web for information discovery.
- As a data mining function, cluster analysis serves as a tool to gain insight into the distribution of data to observe characteristics of each cluster.

13.4 CLASSIFICATION OF VARIOUS CLUSTERING ALGORITHMS

Basically there are two main approaches to clustering the data set. Those two approaches are *partitioning approach* and *hierarchical approach*.

Partitioning method partitions the data set into a predefined set of clusters. Suppose, we have a dataset with **n** objects then partitioning method will partition the dataset to **k** number of clusters. When **k<=n,** it means each cluster will have at least one object or more than one and each object will exactly belong to one group. For a given number of partitions (say k), the partitioning method will create an initial partitioning. Then it uses the iterative re-clustering technique to improve the partitioning by moving objects from one group to other.

Hierarchical method creates a hierarchy of cluster from the data set. **Hierarchical clustering are of two types: *Agglomerative Clustering and Divisive Clustering.*** Agglomerative clustering is a bottom up approach. Divisive clustering technique is a top down approach.

13.4.1 Partitioning Methods

Partitioning clustering algorithm construct partition of a database with N objects to K cluster. The partitioning clustering technique adopts iterative optimization technique. Here, the database is iteratively breaking down to K cluster. Following K-means is a partitioning clustering algorithm.

K-means: K-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed apriori. The main idea is to define k centers, one for each cluster. These centers should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and to associate it to the nearest center. When no point is pending, the first step is completed and an early group age is done. At this point we need to re-calculate ‘k’ new centroids as BabyCenter of the clusters resulting from the previous step. After we have these ‘k’ new centroids, a new binding has to be done between the same data set points and the

nearest new center. A loop has been generated. As a result of this loop we may notice that the k centers change their location step by step until no more changes are done or in other words centers do not move any more. Finally, this algorithm aims at minimizing objective functions known as squared error function given by:

$$J(V) = \sum_{i=1}^{c}\sum_{j=1}^{c_i}(|| x_i - v_j ||)^2$$

where, '$||x_i - v_j||$' is the Euclidean distance between x_i and v_j. 'c_i' is the number of data points in i^{th} cluster. 'c' is the number of cluster centers.

Algorithm 13.1: K-means clustering algorithm

Let $X = \{x_1, x_2, x_3, \ldots, \ldots, ..., x_n\}$ be the set of data points and $V = \{v_1, v_2, \ldots, \ldots, ..., v_c\}$ be the set of centers.

Step 1: Randomly select 'c' cluster centers.

Step 2: Calculate the distance between each data point and cluster centers.

Step 3: Assign the data point to the cluster center whose distance from the cluster center is minimum of all the cluster centers.

Step 4: Recalculate the new cluster center using the formula:

$$v_i = (1/c_i)\sum_{j=1}^{c_i} x_i$$

where, 'c_i' represents the number of data points in i^{th} cluster.

Step 5: Recalculate the distance between each data point and new obtained cluster centers.

Step 6: If no data point was reassigned then stop, otherwise repeat from step 3.



CHECK YOUR PROGRESS

Q.1: State whether the following statements are true (T) or false (F)

i) The main aim of clustering technique is to find both distributed and dense region in a data set.

ii) The partitioning clustering technique does not adopt iterative optimization technique.

iii) The main idea of K mean clustering is to define k centers, one for each cluster.

13.4.2 Hierarchical Methods

Hierarchical method creates a hierarchy of clusters from the data set. Hierarchical clusterings are of two types: ***Agglomerative Clustering and Divisive Clustering.*** Agglomerative clustering is a bottom up approach. In this technique, clustering starts with small cluster where each cluster contains single object and keeps on merging the objects or groups that are close to one another. It keeps on doing so until all of the groups are merged into one or until the termination condition holds. Divisive clustering technique is a top down approach. This technique creates k number of clusters from a single cluster. The following DBSCAN algorithm is a hierarchical algorithm.

Density based DB-SCAN: DBSCAN (Density Based Spatial Clustering of Application of Noise) used a density based concept to discover a new set of cluster. The key idea is that for each object of cluster, the neighborhood of a given radius has to contain a minimum threshold data object. The main parameter of DBSCAN is distance function of data object. Following are some parameter of DBSCAN

ε-Neighbourhood of an object: For a given non negative value ε, the ε-Neighbourhood of an object O_i, denoted by $N_\varepsilon(O_i)$, is defined by $N_\varepsilon(O_i) = \{O_j \varepsilon D \mid d(O_i, O_j) \leq \varepsilon\}$

Core Object: An object is said to be core object if size of the ε-Neighbourhood of the object is greater than mean point. (user specified minimum density) i.e., $| N_\varepsilon (O)| \geq$ MinPts.

Directly Density Reachable: An object O_i is directly density reachable from an object O_j with respect to ε and MinPts, if O_j is a core object and O_i is in its ε-Neighbourhood.

Density Reachable: An object O_i is density reachable from an object O_j with respect to ε and MinPts in D if there is a chain of objects O_1, O_2, O_3, ..., ..., ..., O_n, such that $O_1 = O_j$ and $O_i = O_n$, such that $O_e \varepsilon D$ and O_{e+1} is directly density reachable from O_e.

Density Connected: An object O_i is density connected to an object O_j with respect to ε and MinPts in D if there is another object $O \varepsilon D$ such that both O_i and O_j are density reachable from O.

Cluster: A Cluster C with respect to ε and MinPts is a non empty subset of D satisfying the following conditions:

- For all O_i, $O_j \varepsilon D$, if $O_i \varepsilon C$ and O_j is density reachable from O_i with respect to ε and MinPts, then $O_j \varepsilon C$.
- For all O_i, $O_j \varepsilon C$, O_i is density connected to O_j, with respect to ε and MinPts.

Noise: Let C_1, C_2, ..., ..., ..., C_k are the clusters of D with respect to ε and MinPts. Then the set of objects O are said to be noise if they do not belong to any cluster C_i.

Algorithm 13.2: DBSCAN Algorithm

Input: Database of objects D

Algorithm DBSCAN (D, ε, MinPts)

Step 1: Do for all D
    If O is unclassified
    Call function mk_cluster(O, D, ε, MinPts)
End do

Function mk_cluster(O, D, ε, MinPts)

Step 1: Function mk_cluster(O, D, ε, MinPts)

Step 2: Get the ε-neighbourhood of O as $N_\varepsilon(O)$

Step 3: If $| N_\varepsilon(O)| <$ MinPts
    Mark O as Noise
    return
else

Step 4: Select new cluster_id and mark all object of $N_\varepsilon(O)$ with this cluster_id and put them into candidate objects.

Step 5: do while candidate-objects is not empty

Select an object from candidate-object as current_object

delete current_object from candidate_objects

retrieve N_{ε}(urrent_object)

if | N_{ε}(current_object)| $\geq$ MinPts

select all objects in N_{ε}(current_object) not yet classified or marked as noise,

mark all of the objects with cluster_id,

include the unclassified objects into candidate-objects

end do

return

The DBSCAN algorithm maintains three categories of objects. Those are classified, unclassified and noise. The classified and noise are the objects whose ε-neighbourhood are already calculated. The unclassified objects are those whose ε-neighbourhood is not calculated. Each classified object has an associated cluster-id. Noise has also a dummy cluster-id. The unclassified objects do not have any cluster-id. The algorithm gradually converts the unclassified objects to classified objects and/or noise.



CHECK YOUR PROGRESS

Q.2: Fill in the blanks:

i) Agglomerative clustering is a ..... approach.

ii) Divisive clustering technique is a ..... approach.

iii) An object is said to be core object if size of the ..... of the object is greater than mean point.

iv) The set of objects are said to be noise if they do not belong to any .....



13.5 LET US SUM UP

- Clustering is a usefull technique for the discovery of data distribution and pattern of given data
- The main goal of clustering is to find all sets of similar examples in the data set.
- Two approaches of clustering are partitioning approach and hierarchical approach.
- Partitioning clustering algorithm construct partition of a database with N objects to K cluster.
- K-means is a partitioning clustering algorithm.
- Hierarchical method creates a hierarchy of cluster from the data set.
- DBSCAN(Density Based Spatial Clustering of Application of Noise) used a density based concept to discover a new set of cluster.
- The DBSCAN algorithm maintains three categories of objects, those are classified, unclassified and noise.



13.6 FURTHER READING

1) Han, J., Pei, J. & Kamber, M. (2011). *Data Mining: Concepts and Techniques*. Elsevier.
2) Pujari, A. K. (2001). *Data Mining Techniques*. Universities Press.



13.7 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q. No. 1: i) True; ii) False; iii) True

Ans. to Q. No. 2: i) bottom up; ii) top down; iii) ε-Neighbourhood; iv) cluster



13.8 MODEL QUESTIONS

Q.1: What is clustering technique?

Q.2: What are the different applications of clustering?

Q.3: Explain the various classifications of clustering.

Q.4: Differentiate between Partitioning approach Hierarchical approach.

Q.5: Differentiate between Agglomerative and Divisive clustering techniques.

Q.6: Explain K mean clustering algorithm.

Q.7: What is DBSCAN? Explain the DBSCAN algorithm.

*** ***** ***

UNIT 14: INTRODUCTION TO WEB MINING

UNIT STRUCTURE

14.1 Learning Objectives

14.2 Introduction

14.3 Web Mining

14. 3.1 Types of Web Data

14.4 Categories of Web Data Mining

14.4.1 Web Content Mining

14.4.2 Web Structure Mining

14.4.3 Web Usage Mining

14.5 Applications of Web Data Mining In E-Commerce

14.5.1 Customer Attraction

14.5.2 Customer Retention

14.5.3 Cross-Sales

14.5.4 Improve the E-Commerce Web Site Design

14.6 Text Mining

14.6.1 Process of Text Mining

14.6.2 Applications of Text Mining

14.6.3 Approaches to Text Mining

14.6.4 Mining Result Evaluation of Text Mining

14.7 Episode Rule Discovery for Text

14.8 Let Us Sum Up

14.9 Further Reading

14.10 Answers to Check Your Progress

14.11 Model Questions

14.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- define web mining
- describe the different types of web mining are
- list the uses of web mining
- describe the different methods of web mining

- define text mining
- describe how episode rules are discovered from text.

14.2 INTRODUCTION

In the previous unit, we have learned about the different clustering techniques such as partition algorithm, hierarchical algorithm, divisive algorithm etc. We have also learned about K-means and DBSCAN algorithm in detail. In this unit, we will learn about web mining and its different types. Different categories of web mining along with the applications of web mining are also covered in this unit. Another topic that has been introduced in this unit is text mining. In the next unit, we will explore the concepts of temporal data mining and spatial data mining.

14.3 WEB MINING

Web Mining is the application of data mining techniques to extract useful knowledge from web data like contents of web documents, hyperlinks structure of documents and web usage logs. In web mining, web data refers to the data about web documents, the web structure and web log. The information on web is availed publicly by many companies and persons. Compare to common text documents, the contents of web pages are often complex. The data in the web is very large and it also expands rapidly. Considering all these factors, the WWW has turned into an area of interest for data mining. On WWW, the web documents do not have specific order of arrangement and also it is also a large collection of such documents. According to requirement and interest, these web documents are often updated regularly as the users can have dissimilar background and purpose of usage. It may be possible that only a small portion of information on web is significant and useful to specific user needs. To pursue mining on web all these factors are important and valuable for the discovery and utilization of resources on World Wide Web.

Mining of web data focuses on good design of web sites and formation of techniques to analyse the behaviour of the user. There are also strong

requirements of techniques to help in business decision in e-commerce. By understanding the user behaviour, desired advertising can be displayed on web sites to which the user is likely to respond. Design of web sites is important. A well designed web sites can easily fulfil the objective of business by highlighting the products with good profit margin and its also improves the users navigation experience. Web data mining can be defined in two distinct forms: first, it is defined as chain of order tasks and, second, it is defined considering the type of web data used in web data mining process. There are three types of data generally concerned in web data mining: web contents, web structure and web usage log. According to the kinds of data to be mined, Web Data Mining can be broadly divided into three categories: *Web content mining*, *Web structure mining* and *Web usage mining*.

14.3.1 Types of Web Data

The data on World Wide Web are available in three different formats: web content, web structure and web usage.

- **Web Content Data:** Web content data is the data in the web pages;it can be text, image, audio or video. Web content data are used to provide information to the users. In this category, the HTML pages are common and more familiar form of web content data. Because of variety of internal formats and the browsers way of interpretation, the HTML document may appear differently for different browsers while viewing. The basic document structure of HTML is similar. HTML documents are often considered semi-structured as different elements of documents are not designed according to specific schema.

  XML document is another known form of web content data which enables storing and transporting information. It is having structured information and includes contents as well as information about contents. Each XML document has specific structure and allows identifying document structure and adding information.

  Another type of web content data is dynamic server pages which are processed by the web server and the generated result

is sent to web browser. In contrast, without any change, the static contents are sent to browser. Some of the familiar server page languages are JSP(Java Server Page), ASP(Active Server Page), PHP(Pre-Hypertext Processor) etc.

- **Web Structure data:** Web structure data represents linkage and relationship of web contents to others. Two types of structure, namely *intra-page* and *inter-page* structure can be considered. In specific web page, information about the arrangement of different HTML tags is intra-page structure information. The pages are connected with other pages using hyper-links. This is inter-page structure information. Hyperlinks of web pages collectively form a graph called web graph and it describes the whole structure of the web site. Web graph is a common way of showing the links of one web page to another in WWW and depicts the overall structure. Web graph is a representation of a specific site describing the structure of links and relationship to the HTML documents in WWW.

  Web document is depicted as a node in graph and edge is HTML link connecting one page with another. In two different ways, the edges of the graph are presented. A hyperlink stopping at related page is presented as outgoing arcs and the hyperlinks using which related page can be found is presented as incoming arcs. Web graph can be used in some applications like web searching, indexing and web communities detection.

- **Web Usage data:** Web usage data involves the log data collected by web server and application server. When the user interacts with web site, web log data is generated on web server in form of web server log files. Application Server Data is common in commercial application servers. These data are used to track various types of business events and logs. Application Level Data is another source for web usage data. With this type of data it is possible to record various kinds of events in an application. These data are used for generating histories about selected special events. The data in this category can be divided

into three categories based on the source of its collection: server side data, client side data, and proxy server side data. Other additional data sources are demographic data, site files, cookies etc. Generally, the web server is assigned a domain name and it has IP address. When the user sends request for page to web server through browser, the request is processed by the server and the page is sent to the user. As a result of user interaction with web site and server, data are generated on server and resource request, success, error etc. information are recorded into server log files. Different types of usage log files are created on the server such as access log, error log, referrer log, agent log.

Information stored in web access log includes IP Address, username, date and time, request. Error information like 'file not found', 'no data', 'aborted transmission' etc. is recorded into Error logs. The information about browser, its version and operating system of user making request is stored into Agent logs.

Using web traffic analyser software, the log files of web server can be analysed and useful information can be derived which helps to improve the structure of the web site.

14.4 CATEGORIES OF WEB DATA MINING

Web Data Mining can be divided into three categories according to the kinds of data to be mined.

- Web Content Mining
- Web Structure Mining
- Web Usage Mining

Web content mining is the process of extracting information from the content of web documents on World Wide Web like mining the content of HTML pages. Web structure mining is the process of extracting knowledge from the link structure of the World Wide Web. Web usage mining is the process of discovering interesting patterns from the web access logs on servers. It is also called web log mining. Figure 14.1 represents the categories of web data mining.

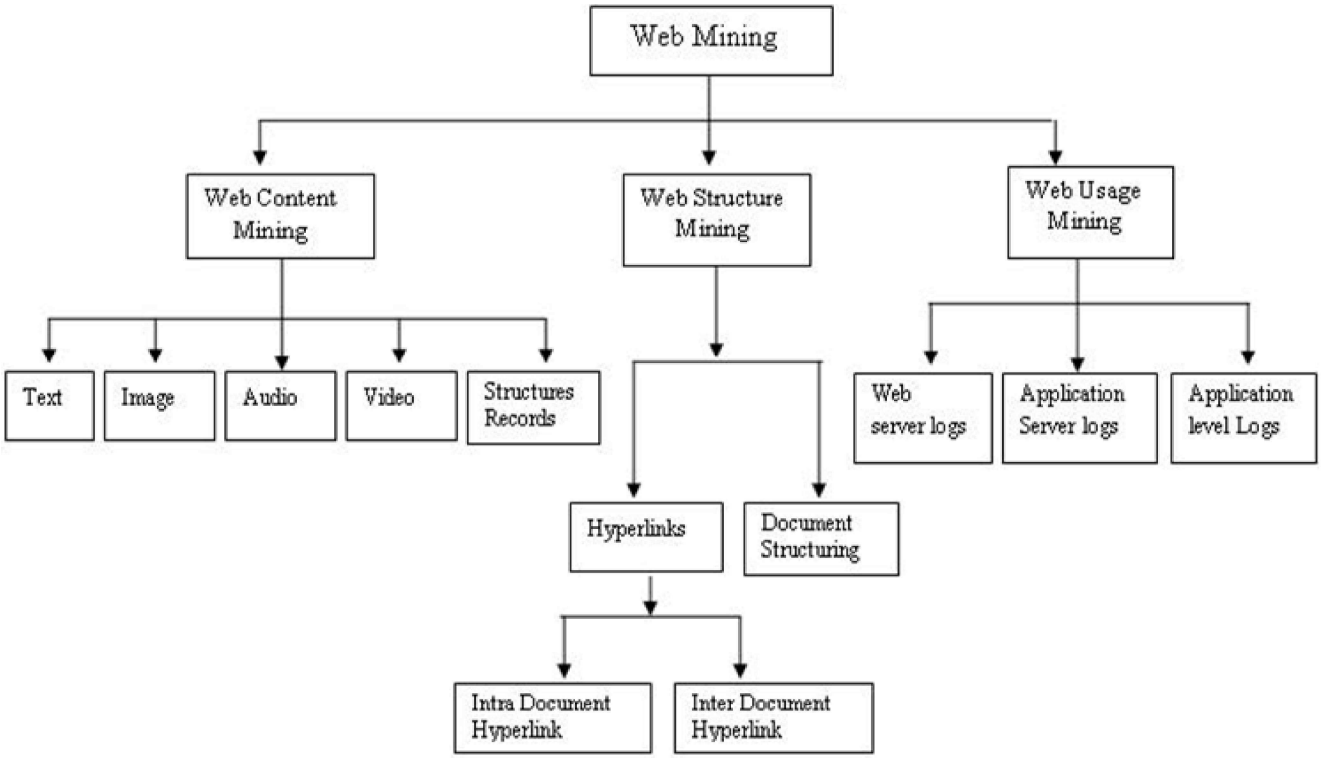


Figure 14.1: Categories of Web Data Mining

14.4.1 Web Content Mining

Web Content Mining is the task of extracting knowledge from the content of documents on World Wide Web. Web Content Mining is a form of text data mining applied to the web domain and has to do with finding the content of documents, classifying documents, and clustering documents. It focuses on finding web documents whose content matches specific criteria. Generally, web document includes the data like text, audio, video, image and hyperlinks. HTML documents are semi structured data. Due to this unstructured nature, web content mining becomes more complex. Web content mining focuses on automatic search of information resources online. The most studied forms of web content mining are text mining and its application including topic discovery, extracting association patterns, clustering and classification of Web Pages.

The multi layered database model was used to transform unstructured data on web into the form relevant to database. To find

relevant documents, intelligent tools are used to extract information from web pages and the abstract information of the relevant documents is contained in the database. Using query language, characterizations are queried from the database. In case of other information, search engine is used to query web resources. The crawler searches the web for documents that match with defined classes. The found documents are saved into database and query techniques are used to query the database.

14.4.2 Web Structure Mining

Web Structure Mining focuses on discovering structure information from the Web to identify the relevant documents. It describes the connectivity in the Web subset based on the given collection of interconnected Web documents. The structure of a typical Web graph consists of Web pages as nodes and hyperlinks as edges connecting the related pages. Mining the site structure and Web page structure can help to guide the classification and clustering of pages to find authoritative pages to improve retrieval performance.

Unlike Web content mining, which mainly concentrates on the information of single document, web structure mining tries to discover the link structures of the hyperlinks between documents. By using information of hyperlinks, web structure mining can classify the Web pages and produce results such as the similarity and relationship between different Web sites. The structural data for Web structure mining is the link between the information and the document structure. Given a collection of web pages and topology, interesting facts related to page connectivity can be discovered. Web document contents can also be represented in a tree-structured format, based on the different HTML and XML tags within the page.

In the area of web structure mining, research on Social Network Analysis has been done. The separation of page is made with respect to the number of incoming and outgoing links. The

methods have been used to calculate the quality and relevance relation of two web pages.

Web structure mining has another dimension, which is to discover the structure of Web document. This dimension of web structure mining is used to extract the schema of Web pages. This information is beneficial for navigation purpose and it provides comparison and integration of Web page schemes. Another task of web structure mining is to discover the nature of the hierarchy or network of hyperlink in the web sites of a particular domain. This may help to generalize the flow of information in Web sites that may represent some particular domain; therefore the query processing can be performed more easily and more efficiently. Web structure mining has a strong relation with the web content mining.

14.4.3 Web Usage Mining

Web Usage Mining is the process of applying data mining techniques to discover interesting patterns from the Web usage data. Web usage mining provides better a understanding for serving the needs of Web-based applications. Web Usage data keeps information about the identity or origin of the Web users with their browsing behaviour in a web domain. Web usage mining itself can be divided into subcategories based on the type of web usage data used. Web server data and application server data are common forms of web usage data.

Most of the research in Web usage mining is focused on applications using web Server Data. Most of the Web log analysis tools use only the textual data from log files. Web usage mining tries to discover useful information from the data extracted from the interactions of the users while surfing on the Web. It also has focus on the methods predicting user behaviour while the user interacts with Web.

The possible application areas of web usage mining are prediction of the user's behaviour within the site, comparison between

expected and actual Web site usage, reconstruction of web site structure based on the interests of its users.

14.5 APPLICATIONS OF WEB DATA MINING IN E-COMMERCE

Using Web mining, deep analysis can be performed by combining other corporate information with Web traffic data. This allows accounting, customer profile, inventory, and demographic information to be correlated with Web browsing that answers complex questions such as:

How many persons purchased something, who visited web site?

Which advertising campaigns resulted in the most purchases?

Do my Web visitors fit a certain profile?

Can I use this for segmenting my market?

Web mining tools can be extended and programmed to answer almost any question. Web mining can provide the companies managerial insight into visitor profiles which help the top management to take strategic actions accordingly. Also, the company can obtain some subjective measurements using Web Mining on the effectiveness of their marketing campaign or marketing research that will help the business to improve and to align their marketing strategies in time.

The company can identify the strength and weakness of its web marketing campaign through Web Mining, and then can make strategic adjustments and obtain the feedback from Web Mining again to see the improvement. This procedure is an on-going continuous process. Some Web Mining applications in e-commerce have been discussed below.

14.5.1 Customer Attraction

The two essential parts of attraction are the selection of prospective new customers and the acquisition of selected potential candidates. One marketing strategy to perform this exercise is to find the common characteristics in already existing visitors' information and behaviour for the classes of profitable and non-profitable customers. These groups are then used as labels for a classifier to

discover Internet marketing rules, which are applied online on site visitors. Depending on the outcome, a dynamically created page is displayed, whose contents depend on the found associations between browser information and the offered products or services.

14.5.2 Customer Retention

Customer retention is the step of managing the process of keeping the online shopper as loyal as possible. Due to the non-existence of physical distances between the providers, this is an extremely challenging task in the electronic commerce scenario. One strategy is similar to the acquisition that is dynamically creating web offers based on associations. However, it has been proved more successful to consider associations across time, also known as sequential patterns. The discovered sequence can be used to display special offers dynamically to keep a customer interested in the site, after a certain page sequence with a threshold support and/or confidence value has been visited.

14.5.3 Cross-Sales

The objective of cross-sales is to diversify selling activities horizontally or vertically to an existing customer base. Traditional generic cross-sales methodology has been adopted in order to perform the given task in an electronic commerce environment. For discovering the potential customers, the characteristic rules of existing cross-sellers have to be discovered. The entire set of discovered rules can then be used as the model to be applied at run-time on incoming actions and requests from existing customers.

14.5.4 Improve the E-Commerce Web Site Design

Attractiveness of the site depends on its reasonable design of content and organizational structure. Web Mining can provide details of the user behaviour, providing web site designers the basis of decision making to improve the design of the site.



CHECK YOUR PROGRESS

Q.1: The main purpose for structure mining is to extract previously unknown relationships between .....

a) Web pages b) Web hyperlinks

c) Web data d) Web contents

Q.2: Web usage mining refers to the discovery of user access patterns from Web usage logs. (Say True / False)

Q.3: Web structure mining is the process of discovering ..... information from the web.

a) Semi structured b) Unstructured

c) Structured d) None of the above

Q.4: The main purpose for content mining is to extract previously unknown relationships of .....

a) Web pages b) Web hyperlinks

c) Web use information d) Web contents

14.6 TEXT MINING

Text mining refers to retrieval of information that is stored in text databases or document databases, which consist of large collections of documents from various sources, such as news articles, research papers, books, digital libraries, e-mail messages, and Web pages. Text databases are growing rapidly due to the increasing amount of information available in electronic form, such as electronic publications, various kinds of electronic documents, e-mail, and the World Wide Web. Now-a-days, most of the information in government, industry, business, and other institutions are stored electronically, in the form of text databases.

The data stored in most text databases are semi structured data in that they are neither completely unstructured nor completely structured. For example, a document may contain a few structured fields, such as title,

authors, publication date, category and so on. But it may also contain some largely unstructured text components, such as abstract and contents.

14.6.1 Process of Text Mining

The process of Text mining involves a series of activities that are performed to mine the information. These activities are shown in the figure 14.2.

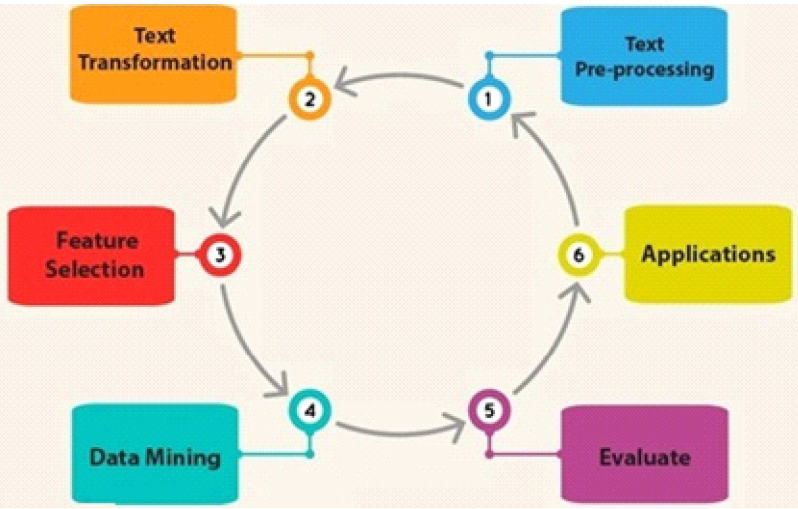


Figure 14.2: Process of Text Mining

Text Pre-processing: It involves a series of steps as shown below:

- **Text Cleanup:** Text Cleanup means removing any unnecessary or unwanted information, such as removing ads from web pages, normalizing text converted from binary formats.
- **Tokenization:** Tokenizing is simply achieved by splitting the text into white spaces.
- **Part of Speech Tagging:** Part-of-Speech (POS) tagging means word class assignment to each token. Its input is given by the tokenized text. Taggers have to cope with unknown words and ambiguous word-tag mappings.

Text Transformation (Attribute Generation): A text document is represented by the words it contains and their occurrences. Two main approaches to document representation are:

i) Bag of words

ii) Vector Space

Feature Selection (Attribute Selection): Feature selection also is known as variable selection. It is the process of selecting a subset of important features for use in model creation by removing the redundant and irrelevant features. Redundant features are the one which provides no extra information. Irrelevant features provide no useful or relevant information in any context.

Data Mining: At this point, the Text mining process merges with the traditional process. Classic Data Mining techniques are used in the text database and model the data as per user requirement

Evaluate: With the help of background knowledge, it is possible to evaluate patterns to find the interesting ones. Normally, a data mining system can discover thousands of patterns. Naturally, many of them are not interesting to the given user, because they represent either common knowledge or lack of novelty. The background knowledge in the form of user-specified constraints or interestingness expectation can be used to guide the discovery process and then reduce the search space.

14.6.2 Applications of Text Mining

Text Mining is applied in a variety of areas. Some of the most common areas are

Risk Management: One of the primary causes of failure in the business sector is the lack of proper or insufficient risk analysis. Adopting and integrating risk management software powered by text mining technologies such as SAS Text Miner can help businesses to stay updated with all the current trends in the business market and also can boost their abilities to mitigate potential risks. Since text mining technologies can gather relevant information from across thousands of text data sources and can create links between the extracted insights, it allows companies to access the right information at the right moment, thereby enhancing the entire risk management process.

Customer Care Service: Text mining techniques, particularly NLP, are finding increasing importance in the field of customer care.

Companies are investing in text analytics software to enhance their overall customer experience by accessing the textual data from varied sources such as surveys, customer feedback, and customer calls, etc. Text analysis aims to reduce the response time of the company and helps address the grievances of the customers speedily and efficiently.

Fraud Detection: Text analytics backed by text mining technologies provides a tremendous opportunity for domains that gather a majority of data in the text format. Insurance and finance companies are harnessing this opportunity. By combining the outcomes of text analyses with relevant structured data these companies are now able to process claims swiftly as well as to detect and prevent frauds.

Business Intelligence: Organisations and business firms have started to leverage text mining techniques as a part of their business intelligence. Apart from providing profound insights into customer behaviour and trends, text mining techniques also help companies to analyse the strengths and weaknesses of their rivals, thus giving them a competitive advantage in the market. Text mining tools such as Cogito Intelligence Platform and IBM text analytics provide insights on the performance of marketing strategies, the latest customer and the market trends etc.

Social Media Analysis: There are many text mining software packages designed exclusively for analysing the performance of social media platforms. These help to track and interpret the texts generated online from the news, blogs, emails, etc. Furthermore, text mining tools can efficiently analyse the number of posts, likes, and followers of our brand on social media, thereby allowing us to understand the reaction of people who are interacting with our brand and online content. The analysis will enable us to understand 'what's hot and what's not' for our target audience.

14.6.3 Approaches to Text Mining

Information Extraction: Information Extraction (IE) refers to the process of extracting meaningful information from vast chunks

of textual data. This method focuses on identifying the extraction of entities, attributes, and their relationships from semi-structured or unstructured texts. Whatever information is extracted is then stored in a database for future access and retrieval. The efficacy and relevancy of the outcomes are checked and evaluated using precision and recall processes.

Information Retrieval: Information Retrieval (IR) refers to the process of extracting relevant and associated patterns based on a specific set of words or phrases. IR systems make use of different algorithms to track and monitor user behaviours and to discover the relevant data accordingly. Google and Yahoo search engines are the two most renowned IR systems.

Categorisation: Text categorisation is a form of "supervised" learning wherein normal language texts are assigned to a predefined set of topics depending upon their content. Thus, categorisation or rather Natural Language Processing (NLP) is a process of gathering text documents and processing and analysing them to uncover the right topics or indexes for each document.

Clustering: Clustering is one of the most crucial techniques of text mining. It seeks to identify the intrinsic structures in textual information and organise them into relevant subgroups or 'clusters' for further analysis. A significant challenge in the clustering process is to form meaningful clusters from the unlabeled textual data without having any prior information on them. Cluster analysis is a standard text mining tool that assists in data distribution or acts as a pre-processing step for other text mining algorithms running on detected clusters.

Summarisation: Text summarisation refers to the process of automatically generating a compressed version of a specific text that holds valuable information for the end user. The aim here is to browse through multiple text sources to craft summaries of texts containing a considerable proportion of information in a concise format, keeping the overall meaning and intent of the original documents essentially the same. Text summarisation integrates and combines the various

methods that employ text categorisation like decision trees, neural networks, regression models, and swarm intelligence.

14.6.4 Mining Result Evaluation of Text Mining

For evaluating the results obtained by using the above techniques for text mining the concept of F measure is used widely. In the following section the concept of F-measure is discussed in detail.

Precision: This is the percentage of retrieved documents that are in fact relevant to the query(i.e., "correct" responses). It is formally defined as:

$$\text{precision} = \frac{|\{\text{Relevant}\} \cap \{\text{Retrieved}\}|}{|\{\text{Retrieved}\}|}$$

Recall: This is the percentage of documents that are relevant to the query and were, in fact,retrieved. It is formally defined as:

$$\text{recall} = \frac{|\{\text{Relevant}\} \cap \{\text{Retrieved}\}|}{|\{\text{Relevant}\}|}$$

F-score, which is defined as the harmonic mean of recall and precision:

$$\text{F_score} = \frac{\text{recall} \times \text{precision}}{(\text{recall} + \text{precision})/2}$$

The harmonic mean discourages a system that sacrifices one measure for another too drastically.

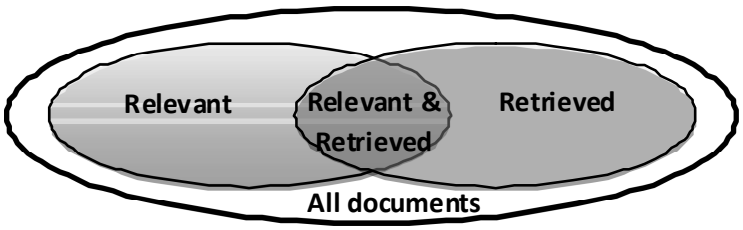


Figure 14.3: Example of relevant and retreived documents

14.7 EPISODE RULE DISCOVERY FOR TEXT

Episode rules and episodes are a modification of the concept of association rules and frequent sets applied to sequential data. Sequential data such as text can be seen as a sequence of pairs where feature vector consists of an ordered set of features and index contains information about

the position of the word in the sequence It is a common practice that the sequence is represented in an increasing order of the indices.

Text episodes are as a pair $\alpha = (V, \leq)$ where V is a collection of feature vectors and $\leq$ is a partial order on V. Given a text sequence S, a text episode $\alpha = (V, \leq)$ occurs within S if there is a way of satisfying the feature vectors in V using the feature vectors in S so that the partial order $\leq$ is respected. Intuitively, this means that the feature vectors of V can be found within S in an order that satisfies the partial order $\leq$.

For an occurrence of the episode to be interesting, all feature vectors of the episode must occur close enough in S. The meaning of close enough is represented as *limit W*, also called as window size. Hence, instead of considering all occurrences of the episode in S, only need to examine the occurrences within substrings S' of S where the difference of the indices of the feature vectors in S' is within the range (close enough). Moreover, since there may be several partially differing occurrences of the episode within the substring S', mining can be restricted only to the distinct minimal occurrences of the episode.

The most useful types of partial orders are– (i) *total orders* i.e., the feature vectors of each episode have a fixed order; such episodes are called serial; and (ii) *trivial partial orders* where the order is not significant at all such episodes are called parallel. A typical example of a serial text episode is a phrase consisting of a sequence of related words with a specific meaning. A typical example of a parallel text episode is a collection of co-occurring terms which may describe the contents of a document better than any of the single term.

The support of α in S is defined as the number of minimal occurrences of α in S. Usually, mining process is interested only in episodes with a support exceeding a given support threshold, meaning that they occur in the sequence frequently enough not to be considered accidental.

An episode rule gives the conditional probability that a certain episode occurs or that a sub-episode has occurred. Formally, an episode rule is an expression $\beta[win_1] = \alpha[win_2]$,where β and α are episodes, β is a sub-episode of α and win_1 and win_2 are window sizes, $win_1 \leq win_2$. The confidence of the rule is the conditional probability that α occurs, given that β occurs under

the window size constraints specified by the rule. An example of an episode rule could be:

knowledge discovery in[4] ⇒ databases[5] (85%)

which tells us that in 85 percent of the cases where the three words "knowledge discovery in" occurred within 4 consequent words, also the word databases occurred within 5 words.



CHECK YOUR PROGRESS

Q.5: In text pre-processing which of the following step is performed?

a) Summarization b) Tokenization

c) Data Extraction d) Data Recovery

Q.6: F-measure is used to evaluate the performance of an IR system (Say True/ False)

Q.7: Which of the following is not a text mining method

a) Summarization b) Clustering

c) Information Retrieval

d) Searching Text in Text Document

Q.8: Application of Text Mining can be found in–

a) Fraud Detection b) Video Analysis

c) Documentation d) None of the above



14.8 LET US SUM UP

- Web Mining is the application of data mining techniques to extract useful knowledge from web data like contents of web documents, hyperlinks structure of documents and web usage logs.
- Mining of web data focuses on well design of web sites and formation of techniques to analyse the behaviour of user. There are also strong requirements of techniques to help in business decision in e-commerce.
- Web Data can be broadly divided into three categories: Web content data, Web structure data and Web usage data.

- Web Data Mining can be divided into three categories according to the kinds of data to be mine: Web Content Mining, Web Structure Mining, Web Usage Mining
- Web content mining is process of extracting information from the content of web documents on World Wide Web like mining the content of HTML pages.
- Web structure mining is the process of extracting knowledge from the link structure of the World Wide Web.
- Web usage mining is the process of discovering interesting patterns from web access logs on servers.
- Text mining refers to retrieval of information is stored in text databases or document database, which consist of large collections of documents from various sources, such as news articles, research papers, books, digital libraries, e-mail messages, and Web pages.
- Data stored in most text databases are semi structured data in that they are neither completely unstructured nor completely structured.
- The process of Text mining involves a series of activities that are performed to mine the information. These activities are: Text Pre-processing, Text Transformation, Feature Selection, Data Mining, Evaluation of results, Application.
- Application of Text mining includes: Risk Management, Customer care service improvement, Fraud Detection, Business Intelligence, Social Media Analysis etc.
- Approaches of text mining includes: Information Extraction, Information Retrieval, Categorisation, Clustering, Summarisation.
- For evaluate the results obtained by using the text mining techniques we use F-measure.
- Episode rules and episodes are a modification of the concept of association rules and frequent sets applied to sequential data.
- Text episodes are as a pair $\alpha = (V, \leq)$ where V is a collection of feature vectors and $\leq$ is a partial order on V.
- An episode rule gives the conditional probability that a certain episode occurs or that a subepisode has occurred.



14.9 FURTHER READING

1) Aggarwal, C. C. (2015). *Data Mining: The Textbook*. Springer.
2) Han, J., Pei, J., & Kamber, M. (2011). *Data Mining: Concepts and Techniques*. Elsevier.
3) Tan, P. N. (2018). *Introduction to Data Mining*. Pearson Education India.



14.10 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q. No. 1: (a) **Ans. to Q. No. 2:** True

Ans. to Q. No. 3: (c) **Ans. to Q. No. 4:** (d)

Ans. to Q. No. 5: (b) **Ans. to Q. No. 6:** True

Ans. to Q. No. 7: (d) **Ans. to Q. No. 8:** (a)



14.11 MODEL QUESTIONS

Q.1: What is Web Mining? What are the different types of web data available for mining? Explain briefly.

Q.2: What are the different types of web mining? Explain briefly.

Q.3: Discuss the different applications of web mining.

Q.4: What is text mining? Explain the different steps of text mining.

Q.5: Explain the different applications of text mining.

Q.6: Explain the different techniques for text mining?

Q.7: What is F-measure? What is meant by Precision and Recall? Explain in detail.

Q.8: What is Episode and Episode rule? Why is episode rule discovery from text important? Explain.

*** ***** ***

UNIT 15: INTRODUCTION TO SPATIAL AND TEMPORAL DATA MINING

UNIT STRUCTURE

15.1 Learning Objectives

15.2 Introduction

15.3 Temporal Data Mining

15.3.1 Types of Data for Temporal Data Mining

15.4 Temporal Data Mining Task

15.5 Temporal Association Rules

15.5.1 Temporal Rules Discovery

15.5.2 Cyclic Association Rules

15.5.3 Calendric Association Rules

15.6 Sequence Mining

15.7 Spatial Data Mining

15.8 Spatial Data Mining Tasks

15.9 Spatial Clustering

15.9.1 Clustering Algorithms Based on Hierarchical Methods

15.9.2 Clustering Algorithms for Partitional Clustering

15.9.3 Clustering Algorithms Based on Density Based Methods

15.9.4 Clustering Algorithms Based on Grid-Based Methods

15.10 Let Us Sum Up

15.11 Further Reading

15.12 Answers to Check Your Progress

15.13 Model Questions

15.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- define temporal data mining
- describe the different temporal data mining tasks
- define temporal association rule mining
- define sequence mining and spatial data mining

- describe the different spatial data mining tasks
- describe spatial clustering algorithms.

15.2 INTRODUCTION

This is the last unit of this course. In the previous unit, we have learned about web mining and text mining. We have also learned about the different categories of web mining along with the applications of web mining. In this unit, we will discuss temporal data mining, its types and different temporal data mining tasks. We will also learn about the different temporal association rules and sequence mining. Spatial data mining and different spatial clustering are also discussed in detail in this unit.

15.3 TEMPORAL DATA MINING

Temporal Data Mining is an important extension to the conventional data mining techniques. *Temporal data mining* is concerned with data mining of large sequential data sets containing the time of occurrence as one of the attributes. In temporal data sets, data are ordered with respect to time. Temporal data mining can be defined as the search for interesting correlations or patterns in large sets of temporal data. It has the capability to discover patterns or rules which might be overlooked when the temporal component is ignored or treated as a simple numeric attribute. By taking into account the time aspect, more interesting patterns are extracted. A large volume of research has, therefore, been focused on temporal data mining to discover temporal rules such as sequential patterns, episodes, temporal association rules and inter transaction association rules, etc.

Time series analysis has quite a long history. Techniques for statistical modelling and spectral analysis of real or complex-valued time series have been in use for more than fifty years. Weather forecasting, financial or stock market prediction and automatic process control have been some of the oldest and most studied applications of such time series analysis. Few people contradict temporal data mining with time series analysis. Following are the few differences between these two.

One main difference between temporal data mining and classical time series analysis lies in the size and nature of data sets and the manner in which the data is collected. Often, temporal data mining methods must be capable of analysing data sets that are prohibitively large for conventional time series modelling techniques to handle efficiently.

The second major difference lies in the kind of information that we want to estimate or unearth from the data. The scope of temporal data mining extends beyond the standard forecast or control applications of time series analysis. Very often, in data mining applications, one does not even know which variables in the data are expected to exhibit any correlations or causal relationships.

Most of the work has been done in the field of temporal data mining which is on single timestamp data, where every piece of information is associated with a timestamp. But not all temporal events can be represented by single timestamp for example, in an e-learning system the user's login and logout time or the user's access time or in a cellular phone company, time and length of each phone call made by the customers, cannot be represented by single timestamps. It requires two timestamp, starting time and ending time of the event or the duration of the event. These types of data are referred to as interval data. Time or duration is not the only attribute that can be described by intervals. There are some other attributes like salary, estimated expenditure, weight, height etc. that can be described by intervals.

Temporal data mining is concerned with data mining of large sequential data sets. By sequential data, it is meant that data is ordered with respect to some index. For example, time series constitute a popular class of sequential data, where records are indexed by time. Other examples of sequential data could be text, gene sequences, protein sequences and lists of moves in a chess game. Although there is no notion of time here as such, the ordering among the records is very important and is central to the data description modelling.

Analysis of temporal time varying data facilitates many interesting challenges. For example, there may be many different interpretations for time. The date stored in the record is the date representing when that

information becomes current. This is often called the valid time. The valid time for information is the time during which the information is true in the modelled world. This usually consists of a start time and an end time. Another time that could have been used is the transaction time. The transaction time is the timestamp associated with the transaction that inserted this record. This could be different from the start time for the valid time interval. The transaction time interval is the time the tuple actually existed in the database. Other types of times may be used as well.

The temporal data often involve the duration of time; that is, a start time and an end time. In this interpretation, the range of values t_s, t_e is associated with each record. Here t_s is the start time and t_e is the end time. A timestamp of a specific time instance may be used instead of a range.

Satellites continually collect images and sensory data. This information is temporal and is associated with specific points in time. When the data were obtained in a hospital, printouts of heartbeats may be kept for patients which represent a continuous view of temporal data. When an electro encephalograph (EEG) is taken for a patient, several different brain waves are measured in parallel. Each wave represents a continuous set of data over time.

Temporal databases usually do not accept the same types of updates and queries as traditional databases. Instead, the only updates that can be allowed are new corrections and versions. Actual modifications of tuple usually are not allowed. Instead, a new tuple with a different valid time would be added.

15.3.1 Types of Data for Temporal Data Mining

Temporal data mining is an important extension of the data mining and it is non-trivial extraction of implicit, potentially useful and previously unrecorded information with an implicit or explicit temporal content, from large database. Data used in temporal data mining can be categorized as following:

Sequences: Sequences are ordered sequence of the events or transaction. Though there may not be any explicit reference to time,

there exists a sort of qualitative temporal relationship between data items. In transactional data a transaction T_i may occur before another transaction T_j or T_i may occur after T_j etc. These types of temporal relationship may be before, after, during, meet and overlap etc. Such relationships are called qualitative relationship between time events.

Time Stamped: This category of the temporal data has explicit time related information. Relationship can be quantitative i.e., we can find the exact temporal distance between data element. The inference made through this type of data may be temporal or non-temporal.

Time Series: Time series data is a special case of the time stamped data. In time series data events have uniform distance on the time scale.

Fully Temporal: Data of this category is fully time dependent. The inferences are also strictly temporal.

15.4 TEMPORAL DATA MINING TASK

The possible objectives of data mining, which are often called tasks of data mining, can be classified into some broad groups. For the case of temporal data mining, these tasks may be grouped as Association rule mining, Prediction, Classification, Clustering, Characterization, Search & retrieval, Pattern discovery, Trend Analysis and Sequence Analysis. These categorization is neither unique nor exhaustive-

1) **Temporal Association Rule Mining:** The discovery of relevant association rules is one of the most important methods used to perform data mining on transactional databases. An effective algorithm to discover association rules is the Apriori. Association rule discovery is an important task in data mining in which we extract the relation among the attribute on the basis of support and confidence. The association rule discovery can be extended to temporal association.

   In general association rules are represented as $X \Rightarrow Y$, which states that if X occurs then Y will occur. The same can be extended to a temporal association rule mining as: $X \Rightarrow (T\ Y)$, which states

that if X occurs then Y will occur within time T. Stating a rule in this new form, enables us to control the impact of the occurrence of an event to the other event occurrence, within a specific time interval.

A cyclic rule is one that occurs at regular time intervals, for example heavy rainfalls occur only in the months of June and July. In order to discover this type of rules, an efficient approach needs to be designed. Discovery of cyclic association rules consists of two processes: first, discover the cyclic large itemsets and second, generate the rules. Extension to this method consists in allowing the existence of different time units, such as days, weeks or months, and is achieved by defining calendar algebra to define and manipulate groups of time intervals. The rules discovered of such type are called cylindrical association rules.

2) **Prediction:** Prediction has a versatile significance in the data mining. It is the forecasting for future on the basis of the past. The task of time-series prediction has to do with forecasting future values of the time series based on its past samples. For prediction a predictive model is designed for the data. There are many nonlinear models for time series prediction such as Neural Networks, neural networks is good for nonlinear modelling of time series data.

   In many cases, prediction may be formulated as classification, association rule finding or clustering problems. Generative models can also be used effectively to predict the evolution of time series. Prediction problems have some specific characteristics that differentiate them from other problems. Prediction gains the importance in various fields like medical, finance, environmental and engineering with an exponential rate.

3) **Classification:** In classification, one classifies the unknown set of attributes in any one of the predefined classes. In temporal classification, each temporal sequence presented in the database is assumed to be belonging to one of the predefined classes or categories and our goal is to automatically determine the corresponding category/class for the given input temporal set of

attributes. There are many examples of sequence classification applications, like handwriting recognition, speech recognition, gesture recognition, demarcating gene and non-gene regions in a genome sequence, on-line signature verification, etc.

4) **Clustering:** Clustering techniques are used to divide the data in the groups on the basis of similarity measure. There exist several clustering technique algorithms like K-means, K-medoids etc. Clustering of sequences or time series is concerned with grouping a collection of time series or sequences based on their similarity. Clustering is of particular interest in temporal data mining since it provides an attractive mechanism to automatically find some structure in large data sets that would be otherwise be difficult to summarize or visualize. There are many applications where a time series clustering activity is relevant e.g., web activity logs. Clusters can indicate navigation patterns of different user groups. Another example could be clustering of biological sequences like proteins or nucleic acids so that sequences within a group have similar functional properties.

5) **Characterization:** Characterization is nothing but summarization of the general characteristics or features of a target class of data. Characterization can be extended to temporal data. An interesting experiment would be extending to the concept of decision tree construction on temporal attributes. For example a rule could be: the first case of filarial is normally reported after the first pre-monsoon rain and during the month of May-August. The output of characterization can be represented in various forms e.g., pie charts, bar charts, curves, multidimensional data cubes.

6) **Search and Retrieval:** The problem of searching is concerned with efficiently locating subsequences often referred to as queries in large archives of sequences or sometimes in a single long sequence. Query-based searches have been extensively studied in language and automata theory. However the problem of efficiently locating exact matches of substrings is well solved, the situation is different

when looking for approximate matches. In content-based retrieval, a query is presented to the system in the form of a sequence. The task is to search a typically large data base of sequential data and retrieve from it sequences or subsequences similar to the query sequence.

Essentially we need to properly insert 'gaps' in the two sequences or decide which should be corresponding elements in the two sequences. Time warping methods have been used for sequence classification and matching for many years. In speech applications, Dynamic Time Warping (DTW) is a systematic and efficient method based on dynamic programming that identifies which correspondence among the feature vectors of two sequences is the best when scoring the similarity between them. In recent times, there are many situations in which sequence matching are used. For example, many biological sequences such as genes, proteins, etc.

7) **Pattern Discovery:** Unlike in search and retrieval applications, in pattern discovery there is no specific query in hand with which to search the database. The objective is simply to unearth all the patterns of interest. Sequence prediction, Classification, Clustering and Search and retrieval methods originally come from other disciplines like estimation theory, machine learning or pattern recognition. On the other hand, the pattern discovery task has its origins in data mining itself. Pattern discovery is exploratory and an unsupervised nature of operation, which is something of a sole preserve of data mining.

8) **Trend Analysis:** Trend analysis in temporal database is referred to change in attribute due to the change in time. The analysis of one or more time series of continuous data may show similar trends, i.e. similar shape across the time axis. Here we are trying to find the relationships of change in one or more static attributes, with respect to changes in the temporal attributes. Trend analysis is very useful for the decision support system and the decision maker.

15.5 TEMPORAL ASSOCIATION RULES

The problem of the discovery of association rules comes from the need to discover patterns in transaction data in a supermarket. But transaction data in supermarkets are temporal. When gathering data about products purchased in a supermarket, the time of the purchase is registered in the transaction. In large volume temporal data, as used for data mining purposes, information may be found related to products that did not necessarily exist throughout the data collection period. So some, products may be found which, at the moment of mining are already discontinued. There may be also new products that were introduced after the beginning of the data collection. Some of these new products will participate in the associations, but may not be included in association rule as support count of such item is not up to the mark. For example, if the total number of transactions is 3,000,000 and we fix the minimum support as 0.5% which is 150,000. Let a particular product has been sold during the last 30 months and has just the minimum support i.e. it appears on an average 5,000 transactions per month. Now consider another product that was incorporated in the last 6 months and that appears in 20,000 transactions per month. The total number of transactions in which it occurs is 120, 000 so the product is infrequent as it has not attained the minimum support threshold even if it is four times as popular as the first one. However, if we consider the introduction time of the product in the market, its support may be more than the minimum support threshold. So, by introduction of time aspect to traditional association, rule mining is more interesting and potentially useful association rules can be mined.

One more example can be when large volumes of data from supermarket are used for mining. There may be information related to certain products that have not existed throughout the dataset but they appear only in certain period. If we mine such datasets without considering the time aspect we may miss out some interesting association rules. For example, cold drinks are specially sold in the summer and if we consider the whole dataset without considering the time aspect, cold drink may be participated in any of the association rules as it has not attained the minimum support threshold.

One way to solve this problem is by incorporating time in the model of discovery of association rules. We will call these rules Temporal Association Rules. A temporal association rule can be defined as a pair (R, T) where R is an association rule and T is a temporal feature, such as a period or a calendar.

One sub product of this idea is the possibility of eliminating outdated rules, according to the user criteria. Moreover, it is possible to delete obsolete itemsets by considering the lifetime of the itemset which will reduce the amount of work to be done in the determination of the frequent itemsets and hence, in the determination of the rules.

The temporal association rules are the extension of the non-temporal model. The basic idea was to limit the search for frequent sets of items or itemsets to the lifetime of the itemset's members. For that reason, the concept of temporal support was introduced. Thus, each rule has an associated timeframe, corresponding to the lifetime of the items participating in the rule. If the extent of a rule's lifetime exceeds a minimum stipulated time set by the user, then the rule is analysed for its validity in that period. This concept allows us to find rules with the traditional frequency counting algorithms.

15.5.1 Temporal Rules Discovery

*The discovery of all the temporal association rules in a transaction set **D** can be made in two phases*

Phase 1: Finding of all the frequent itemsets in their lifespan i.e. the itemsets whose frequency exceeds the user's specified minimum support σ and temporal support.

Phase 2: Using the above frequent sets to find the association rules.

15.5.2 Cyclic Association Rules

If the association rules are computed over monthly sales data, some seasonal variation may be observed where certain rules are true approximately in the same month each year. Similarly, association rules can also display regular hourly, daily, weekly, etc. which are called cyclic association rules.

15.5.3 Calendric Association Rules

If the temporal association rules with time intervals which follow some user-given calendar schema then it is called calendric association rules. An example of such schema is (year, month, day), which yields a set of calendar based patterns. Calendric association rules are repeated with the calendar schemas.



CHECK YOUR PROGRESS

Q.1: State whether the following statements are True or False:

a) Temporal Data Mining is an important extension of conventional data mining techniques.

b) Temporal data mining and classical time series analysis are same.

c) Temporal association rules are represented as $X \Rightarrow (T\ Y)$.

d) Pattern discovery is an temporal data mining task.

Q.2: Which of the following dataset is not temporal?

a) Sequence dataset b) Interval dataset

c) Time series dataset d) None of the above

Q.3: Which of the following is not a temporal data mining task?

a) Classification b) Prediction

c) Select data from dataset d) Pattern Discovery

15.6 SEQUENCE MINING

Sequence mining is a data mining task specialized for analysing sequential data, to discover sequential patterns. More precisely, it consists of discovering interesting sub-sequences in a set of sequences, where the interestingness of a subsequence can be measured in terms of various criteria such as its occurrence frequency, length or profit. Sequence mining has numerous real-life applications due to the fact that data is naturally encoded as sequences of symbols in many fields such as bioinformatics, e-learning, market basket analysis, texts, and webpage click-stream analysis.

Let us discuss the sequence mining with an example. Consider the following sequence database, representing the click/opening of web pages by users.

SID	Sequences
1	<{a,b}, {c}, {f,g}, {g}, {e}>
2	<{a,d}, {c}, {b}, {a,b,e,f}>
3	<{a}, {b}, {f,g}, {e}>
4	<{b}, {f,g}>

This database contains four sequences. Each sequence represents the click stream by users at different times. A sequence is an ordered list of click streams. For example, in this database, the first sequence (SID 1) indicates that an user opens the webpage *a* and then opens page *b* by following hyperlink from page *a*, then in some other time opens the webpage *c*, then opens webpage *f* and *g*, then opens page *g*, and then finally open page *e* in some other time.

Traditionally, sequence mining is being used to find sub-sequences that appear often in a sequence database, i.e., that are common to several sequences. Those sub-sequences are called the frequent sequential patterns. For example, in the context of our example, sequence mining can be used to find the sequences of click-streams that is followed by most of the users. This can be useful to understand the behaviour of users and to arrange link to different webpages so that a user can find those webpages easily.

To perform sequence mining, a user must provide a sequence database and specify a parameter called the minimum support threshold. This parameter indicates a minimum number of sequences in which a pattern must appear to be considered frequent, and be shown to the user. For example, if a user sets the minimum support threshold to 2 sequences, the task of sequence mining consists of finding all sub-sequences appearing in at least 2 sequences of the input database. In the example database, 30 sub-sequences met this requirement. These sequential patterns are shown in the table below, where the number of sequences containing each pattern (called the *support*) is indicated in the right column of the table as given below.

Pattern	Support
<{a}>	3
<{a}, {g}>	2
<{a}, {g}, {e}>	2
<{a}, {f}>	3
<{a}, {f}, {e}>	2
<{a}, {c}>	2
<{a}, {c}, {f}>	2
<{a}, {c}, {e}>	2
<{a}, {b}>	2
<{a}, {b}, {f}>	2
<{a}, {b}, {e}>	2
<{a}, {e}>	3
<{a}, {f, g}>	2
<{a, b}>	2
<{b}>	4
<{b}, {g}>	3
<{b}, {g}, {e}>	2
<{b}, {f}>	4
<{b}, {f, g}>	3
<{b}, {f}, {e}>	2
<{b}, {e}>	3
<{c}>	2
<{c}, {f}>	2
<{c}, {e}>	2
<{e}>	3
<{f}>	4
<{f, g}>	3
<{f}, {e}>	2
<{g}>	3
<{g}, {e}>	2

For example, the patterns <{a}> and <{a}, {g}> are frequent and have a support 3 and 2 respectively. In other words, these patterns appear

in 3 and 2 sequences of the input database. The pattern <{a}> appears in the sequences 1, 2 and 3, while the pattern <{a}, {g}> appears in sequences 1 and 3. These patterns are interesting as they represent some behaviour common to several users. In real life sequence mining can actually be applied on database containing hundreds of thousands of sequences.

Another example of application of sequence mining is text analysis. In this context, a set of sentences from a text can be viewed as sequence database, and the goal of sequence mining is then to find sub-sequences of words frequently used in the text. If such sequences are contiguous, they are called “n-grams”.

15.7 SPATIAL DATA MINING

Spatial data mining is the application of data mining to spatial models. In spatial data mining, analysts use geographical or spatial information to produce business intelligence or other results. This requires specific techniques and resources to get the geographical data into relevant and useful formats.

Challenges involved in spatial data mining include identifying patterns or finding objects that are relevant to the questions that drive the research project. Analysts may be looking in a large database field or other extremely large data set in order to find just the relevant data, using GIS/GPS tools or similar systems.

One interesting thing about the term “spatial data mining” is that it is generally used to talk about finding useful and non-trivial patterns in data. The core goal of a spatial data mining project is to distinguish the information in order to build real, actionable patterns to present, excluding things like statistical coincidence, randomized spatial modelling or irrelevant results. One way analysts may do this is by combing through data looking for “same-object” or “object-equivalent” models to provide accurate comparisons of different geographic locations.

Spatial data mining is a non-trivial process to extract knowledge, spatial relationships, or other interesting patterns not explicitly stored in spatial databases. For spatial data mining it is required to integrate data

mining with spatial database technologies. It can be used to understand spatial data, relationships between spatial and non-spatial data, discovery of spatial relationships, constructing spatial knowledge bases, reorganizing spatial databases, and optimizing spatial queries. It has a wide application in geographic information systems (GIS), vector and raster image database, geo-marketing, remote sensing, image database exploration, environmental studies, and many other areas where spatial data are used. However, in comparison to extracting patterns from traditional numeric and characterized data, it is much difficult to extract interesting and useful patterns from spatial databases because of the complexity of spatial data types, spatial relationships, and spatial autocorrelation.

A huge volume of data is being generated with the growing production of maps which exceed people's capacity to analyse them. Voluminous geographic data have been, and continue to be, collected with modern data acquisition techniques such as global positioning systems (GPS), high-resolution remote sensing, location-aware services and surveys, and internet-based volunteered geographic information. Therefore the knowledge discovery methods like data mining can appropriately be applied to spatial data. This recent concept is an extension of the conventional data mining tasks and is applied to alphanumerical and spatial data. However in spatial data mining the spatial analysis takes into account spatial relations between objects. During the last few years, due to the widespread applications of GPS technology, web-based spatial data sharing and mapping, high-resolution remote sensing, and location-based services, more and more research domains have created or gained access to high-quality geographic data to incorporate spatial information and analysis in various studies, such as social analysis and business applications.

Spatial data mining is becoming quite popular as there are many applicative areas such as ecology, transportation, epidemiology etc, where the concept is being utilized. There are many diversified approaches and algorithms that have been proposed to extract knowledge from such data. However the main challenge to apply any of the algorithms and how to automate the data preparation task, which consumes most of the effort and

time required for knowledge discovery in geographic databases. In spatial databases data are stored in different relations and it is required to join them spatially for finding out novel and useful pattern of data.

Since the conventional data mining methods are not suited for spatial data as they neither support location data nor the implicit relationships between spatial objects. Therefore, it is needed to develop some new methods of spatial data mining to find spatial relationships. Calculating the spatial relationship is a time consuming task in spatial database because the data is generated by encoding geometric location. Because of this complexity the global performances will suffer.

However, it is highly desired that the existing methods of data mining are extended and incorporated into spatial data mining methods. Spatial and GIS methods are crucial for spatial join, data access and graphical map display. The conventional data mining methods are able to generate knowledge about alphanumerical properties only.

15.8 SPATIAL DATA MINING TASKS

Spatial data mining tasks are the extension of conventional data mining tasks where spatial data and criteria are combined. The following tasks are performed in spatial data mining:

i) Spatial and non-spatial data are summarized,
ii) Deviations detection is done after looking for general trends,
iii) Classification rules are discovered,
iv) Clusters of similar objects are formed and
v) Associations and dependencies identified to characterize data.

To carry out these tasks different methods are used. Some of these methods are derived from statistics and others from the field of machine learning.

i) DATA SUMMARIZATION: It is the most common approach in which we apply elementary statistics, such as the calculation of mean, average, variance, histogram and pie chart formation etc. To establish spatial relationship between the objects we use the concept of contiguity matrix. Contrary to the autocorrelation measure, density

analysis forms part of Exploratory Spatial Data Analysis (ESDA), which does not require any knowledge about data. In this the non-spatial properties are ignored. In geographic data analysis both the alphanumerical property data and spatial data are taken into consideration.

ii) **GENERALISATION:** In this method, abstract level of non-spatial attributes is raised and the details of geometric description are reduced by merging adjacent objects. It is derived from attribute-oriented induction concept. The concept hierarchy is the important representation which can be spatial or the thematic map of non-spatial data. Generalizations are of two types:

1) Non-spatial dominant generalization: In this the thematic hierarchy is used first and then adjacent objects are merged.
2) Spatial dominant generalization: This generalization is based on a spatial hierarchy. After this the aggregation or generalization of non-spatial values for each generalized spatial value is formed.

iii) **CLASSIFICATION:** Classification is about grouping data items into classes or categories according to their properties. Classification is also called supervised learning, it needs a training dataset to train the classification model, a testing dataset is used to evaluate the performance of the trained model. Classification methods include, decision trees, artificial neural networks (ANN), maximum likelihood estimation (MLE), linear discriminant function (LDF), support vector machines (SVM), nearest neighbour methods and case-based reasoning (CBR).

iv) **CLUSTERING:** Cluster analysis is mainly used for data analysis. It organizes a set of data items into groups known as clusters. The items in the same cluster are similar to each other and different from those in other clusters. Many different clustering methods have been developed in various research fields such as statistics, pattern recognition, data mining, machine learning, and spatial analysis.

To consider spatial information in clustering, three types of clustering analysis have been studied, spatial clustering,

regionalization, and point pattern analysis. For the first type, spatial clustering, the similarity between data points or clusters is defined with spatial properties. Spatial clustering methods can be partitioning or hierarchical, density-based, or grid-based.

v) **SPATIAL DATA DEPENDENCIES:** It is the local autocorrelation method to work with spatial data which reflects how data are related. Spatial association rule, which works on spatial data dependence, have been adapted to spatial data. Spatial auto-correlation is concerned with the assessment of the degree of spatial dependence. This is done by using the concept of spatial weight matrix. It is equivalent to a residual test in regression analysis. This makes it possible to measure the difference between the actual spatial distribution of variable values and a random one.

15.9 SPATIAL CLUSTERING

Spatial clustering is a process of grouping a set of spatial objects into groups called clusters. Objects within a cluster show a high degree of similarity, whereas the clusters are as much dissimilar as possible. Outlier is a data point that does not conform to the normal points characterizing the data set. Detecting outlier has important applications in data mining as well as in the mining of abnormal points for fraud detection, stock market analysis, intrusion detection, marketing etc.

Important data characteristics, which affect the efficiency of clustering techniques, include the type of data (nominal, ordinal, numeric), data dimensionality (since some techniques perform better in low-dimensional spaces) and error (since some techniques are sensitive to noise).

Clustering of points is a most typical task in spatial clustering, and many kinds of spatial objects clustering can be abstracted as or transformed to points clustering. Here we will discuss about various kinds of spatial clustering. Spatial clustering is the most complicated among all the spatial data mining techniques.

Based on the technique adopted to define clusters, the clustering algorithms can be divided into four broad categories:

a) Hierarchical clustering methods (AGNES, BIRCH etc.),
b) Partitional clustering algorithms (K-means, K-medoids etc.),
c) Density-based clustering algorithms (DBSCAN DENCLUE etc.),
d) Grid based clustering algorithms (STING etc.).

Many of these can be adapted to or are specially tailored for spatial data.

15.9.1 Clustering Algorithms Based on Hierarchical Methods

Hierarchical clustering is a conventional clustering method which has wide variety of applications in different domains. Mainly, it is of two types:

a) Agglomerative Hierarchical clustering algorithm or AGNES (agglomerative nesting) and
b) Divisive Hierarchical clustering algorithm or DIANA (divisive analysis).

Both this algorithm are exactly reverse of each other.

Agglomerative Nesting (AGNES): AGNES proceeds by a series of fusions of similar objects. Initially all objects are apart– each object forms a small cluster by itself. At the first step, two closest or minimally dissimilar objects are merged to form a cluster. Then the algorithm finds a pair of objects with minimal dissimilarity. If there are several pairs with minimal dissimilarity, the algorithm picks a pair of objects at random. Picking of pairs of objects, which are minimally dissimilar. For finding out the minimally dissimilar objects dissimilarity measures are used. AGNES computes Agglomerative Coefficient (AC), which measures the clustering structure of the data set.

Divisive Analysis (DIANA): DIANA is a hierarchical clustering technique, but its main difference with the agglomerative method (AGNES) is that it constructs the hierarchy in the inverse order. Initially, there is one large cluster consisting of all n objects. At each subsequent step, the largest available cluster is split into two clusters until finally all clusters, comprise of single objects. Thus, the hierarchy

is built in n-1 steps. In the first step of an agglomerative method, all possible fusions of two objects are considered leading to combinations. In the divisive method based on the same principle, there are possibilities to split the data into two clusters. This number is considerably larger than that in the case of an agglomerative method.

When applying AGNES and DIANA to the spatial data, each object in the data set is replaced by spatial points. These spatial points are similar to those in is the 2D plane with x and y coordinates. The main difference is that spatial points are characterized by latitude and longitude. The latitude is the location of a place on the earth, north or south of the equator and longitude is the east – west measurement of position on the earth, measured from a plane running through polar axis. Here, the similarity between objects is expressed in terms of the Euclidian distance between two points in the n-dimension.

15.9.2 Clustering Algorithms for Partitional Clustering

Partitional clustering is another kind of conventional clustering method which decomposes a data set into a set of disjoint clusters. Given a data set of N points, a partitioning method constructs K (N e" K) partitions of the data, with each partition representing a cluster. That is, it classifies the data into K groups by satisfying the following requirements:

1) each group contains at least one point, and
2) each point belongs to exactly one group.

Notice that for fuzzy partitioning, a point can belong to more than one group.

K-Means Algorithm: K-means (MacQueen, 1967) is one of the simplest unsupervised learning algorithms that solve the well-known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids should be placed in a cunning way because of

different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early grouping is done. At this point, re-calculate k new centroids of the clusters resulting from the previous step. After having these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. This procedure has to repeat until there is no change in the sets of centroids between two iterations.

K-Medoid Algorithm: The k-medoid algorithm is a clustering algorithm related to the k-means algorithm. Both the k-means and k-medoid algorithms are partitional and both attempt to minimize the distance between points labelled to be in a cluster and a point designated as the centre of that cluster. In contrast to the k-means algorithm, k-medoids chooses datapoints as centres and works with an arbitrary matrix of distances between data points. A medoid can be defined as the object of a cluster, whose average dissimilarity to all the objects in the cluster is minimal. In some applications, we need each centre to be one of the points itself. This is where K-medoid comes in.

Even though these algorithms can be used for spatial clustering, but the main disadvantage is that these methods are sensitive to the noise and centre point. Another disadvantage is that these can detect only spherical clusters. So, we prefer density based clustering methods for spatial related applications than hierarchical and partitional.

15.9.3 Clustering Algorithms Based on Density Based Methods

Density-based approaches apply a local cluster criterion. Clusters are regarded as regions in the data space in which the objects are dense, and which are separated by regions of low object

density (noise). These regions may have an arbitrary shape and the points inside a region may be arbitrarily distributed. Density based clustering algorithm includes DBSCAN, DENCLUE, OPTICS etc.

DBSCAN: DBSCAN requires two parameters: ε (eps) and the minimum number of points required to form a cluster (minPts). It starts with an arbitrary starting point that has not been visited. This point's ε-neighbourhood is retrieved, and if it contains sufficiently many points, a cluster is started. Otherwise, the point is labelled as noise. Note that this point might later be found in a sufficiently sized ε-environment of a different point and hence is made part of a cluster. If a point is found to be a dense part of a cluster, its ε-neighbourhood is also part of that cluster. Hence, all points that are found within the ε-neighbourhood are added. This process continues until the density-connected cluster is completely found. Then, a new unvisited point is retrieved and processed, leading to the discovery of a further cluster or noise.

DENCLUE: DENCLUE (DENsity basted CLUstEring) is a generalization of partitioning, locality based and hierarchical or grid-based clustering approaches. The algorithm models the overall point density analytically using the sum of the influence functions of the points. Determining the density-attractors causes the clusters to be identified. DENCLUE can handle clusters of arbitrary shape using an equation based on the overall density function.

The approach of DENCLUE is based on the concept that the influence of each data point on its neighbourhood can be modelled mathematically. The mathematical function used, is called an impact function. This impact function is applied to each data point and the density of the data space is the sum of the influence function for all the data points. In DENCLUE, since many data points do not contribute to the impact function, local density functions are used. Local density functions are defined by a distance function like Euclidean distance.

The local density functions consider only data points which actually contribute to the impact function. Local maxima, or density-

attractors identify clusters. These can be either centre-defined clusters, similar to k-means clusters, or multi-centre-defined clusters, that is a series of centre-defined clusters linked by a particular path. Multi-centre-defined clusters identify clusters of arbitrary shape. Clusters of arbitrary shape can also be defined mathematically.

15.9.4 Clustering Algorithms Based on Grid-Based Methods

Grid based methods quantize the object space into a finite number of cells (hyper-rectangles) and then perform the required operations on the quantized space. The main advantage of Grid based method is its fast processing time which depends on number of cells in each dimension in quantized space. Some grid based methods are CLIQUE (CLustering In QUEst), STING (STatistical INformation Grid), Wave clustering etc.

STING (STatistical INformation Grid based method): This is a grid based multi resolution clustering technique in which the spatial area is divided into rectangular cells(using latitude and longitude) and employs a hierarchical structure. Corresponding to different resolution, different levels of rectangular cells are arranged to form a hierarchical structure. Each cell at a higher level is partitioned into a number of cells at its immediate lower level and so on. Statistical information associated with each cell is calculated and is used to answer queries.

Wave Clustering: Wave Cluster is a multi-resolution clustering algorithm. It is used to find clusters in very large spatial databases. Given a set of spatial objects O_i, $1 \leq i \leq N$, the goal of the algorithm is to detect clusters. It first summarizes the data by imposing a multi-dimensional grid structure on to the data space. The main idea is to transform the original feature by applying wavelet transform and then find the dense regions in the new space. A wavelet transform is a signal processing technique that decomposes a signal into different frequency sub bands.

CLIQUE (CLustering In QUEst): The CLIQUE algorithm integrates density based and grid based clustering unlike other clustering algorithms described earlier, CLIQUE is able to discover clusters in the subspace of the data. It is useful for clustering high dimensional data which are usually very sparse and do not form clusters in the full dimensional space.

In CLIQUE, the data space is partitioned into non-overlapping rectangular units by equal space partition along each dimension. A unit is dense if the fraction of total data points contained in it exceeds an input model parameter.

CLIQUE performs multidimensional clustering by moving from the lower dimensional space to higher. When it searches for dense units at the k-dimensional space, CLIQUE make use of information that is obtained from clustering at the (k–1) dimensional space to prune off unnecessary search.

CLIQUE automatically finds subspace of highest dimensionality such that high density clusters exist in those subspaces. It is intensive to the order of input tuples and does not presume any canonical distribution. It scales linearly with the size of input and has good scalability as the number of dimensions in the data is increased. However, the accuracy of clustering results may be degraded at the expense of the simplicity of the method.

The spatial clustering algorithms are categorized based on four factors like time complexity, input, handling of higher dimensional data, capability to detect irregularly shaped clusters. These factors were found to be necessary for effective clustering of large spatial datasets with high dimensionality. The previous nine algorithms are addressed by how they matched these four requirements. It can be seen that although several of the algorithms meet some of the requirements, and some meet most, none of the algorithms as originally proposed meet all the requirements. The hierarchical clustering methods are similar in performance but consume more time as compared to the other. The performance of partitional

clustering methods like K-means and K-medoid algorithms are not well in handling irregularly shaped clusters. The density based methods and grid based methods are more suitable for handling spatial data, but when considering time complexity, grid based methods are more preferable.



CHECK YOUR PROGRESS

Q.4: Which of the following is a density based clustering algorithm

a) CLIQUE b) DBSCAN

c) K-Means d) None of the above

Q.5: Which of the following is not a partitional clustering algorithm?

a) K-means b) K-Medoids

c) AGNES d)

Q.6: STING is a _____.

a) Hierarchical Clustering Algorithm

b) Partitional Clustering Algorithm

c) Density Based Clustering Algorithm

d) Grid-Based Clustering Algorithm

Q.7: Which of the following is not a spatial data mining task?

a) Data summarization b) Generalization

c) Classification d) Association Rule Mining



15.10 LET US SUM UP

- Temporal Data Mining is an important extension to the conventional data mining techniques.
- In temporal data sets data are ordered with respect to time.
- Temporal data mining can be defined as the search for interesting correlations or patterns in large sets of temporal data.
- Data used in temporal data mining can be categorized as– Sequences, Time Stamped, Time Series, Fully Temporal data.

- Temporal data mining tasks can be categorized as Association rule mining, Prediction, Classification, Clustering, Characterization, Search & retrieval, Pattern discovery, Trend Analysis and Sequence Analysis.
- A temporal association rule is of the form $X \Rightarrow (T\ Y)$, which states that if X occurs then Y will occur within time T.
- A cyclic rule is one that occurs at regular time intervals
- If the temporal association rules with time intervals which follow some user-given calendar schema then it is called calendric association rules.
- Sequence mining is a data mining task specialized for analysing sequential data, to discover sequential patterns.
- Traditionally, sequence mining is being used to find sub-sequences that appear often in a sequence database.
- Spatial data mining is the application of data mining to spatial models.
- In spatial data mining, analysts use geographical or spatial information to produce business intelligence or other results.
- Spatial data mining is a non-trivial process to extract knowledge, spatial relationships, or other interesting patterns not explicitly stored in spatial databases.
- Spatial data mining tasks are the extension of conventional data mining tasks where spatial data and criteria are combined.
- Spatial data mining tasks are can be categorized as- Data Summarization, Generalisation, Classification, Clustering, Spatial Data Dependencies,
- Spatial clustering is a process of grouping a set of spatial objects into groups called clusters.
- Based on the technique adopted to define clusters, the clustering algorithms can be divided into four broad categories:
  - Hierarchical clustering methods (AGNES, BIRCH etc.),
  - Partitional clustering algorithms (K-means, K-medoids etc.),
  - Density-based clustering algorithms (DBSCAN DENCLUE etc.),
  - Grid based clustering algorithms (STING etc.).



15.11 FURTHER READING

1) Aggarwal, C. C. (2015). Data Mining: The Textbook. Springer.
2) Han, J., Pei, J., & Kamber, M. (2011). *Data Mining: Concepts and Techniques*. Elsevier.
3) Tan, P. N. (2018). Introduction to Data Mining. Pearson Education India.



15.12 ANSWERS TO CHECK YOUR PROGRESS

Ans. to Q. No. 1: a) True; b) False; c) True; d) True

Ans. to Q. No. 2: (d) **Ans. to Q. No. 3:** (c)

Ans. to Q. No. 4: (b) **Ans. to Q. No. 5:** (c)

Ans. to Q. No. 6: (a) **Ans. to Q. No. 7:** (d)



15.13 MODEL QUESTIONS

Q.1: What is temporal data mining? Explain the different types of temporal data.

Q.2: Explain the different temporal data mining tasks.

Q.3: What is temporal association rule mining? What are the different types of temporal association rules? Explain.

Q.4: What is sequence mining? Explain with example.

Q.5: Explain the different spatial data mining task.

Q.6: What is spatial data mining? Explain briefly.

Q.7: What is spatial data clustering? What are the different types of Spatial data clustering? Explain in detail.

*** ***** ***



REFERENCES

1) Aggarwal, C. C. (2015). *Data Mining: The Textbook*. Springer.
2) Berson, A., & Smith, S. J. (1997). *Data Warehousing, Data Mining and OLAP*. McGraw-Hill, Inc.
3) Han, J., Pei, J. & Kamber, M. (2011). *Data Mining: Concepts and Techniques*. Elsevier.
4) Inmon, W. H. (2005). Building the Data Warehouse. John Wiley & Sons.
5) Ponniah, P. (2004). *Data Warehousing Fundamentals: A Comprehensive Guide for IT Professionals*. John Wiley & Sons.
6) Pudi, V., Krishna R. P. (2008) *Data Mining*. Oxford University Press.
7) Pujari, A. K. (2001). *Data Mining Techniques.* Universities Press.
8) Saxena, A., Saxena, K. Saxena, S. (2015). *Data Mining and Warehousing.* BPB Publications.
9) Tan, P. N. (2018). *Introduction to Data Mining*. Pearson Education India.
10) Tan, P. N., Steinbach, M. & Kumar, V. (2013). *Data Mining Cluster Analysis: Basic Concepts and Algorithms. Introduction to Data Mining*.
11) https://www.tutorialspoint.com/data_mining/dm_quick_guide.htm

Name of the Paper: .....
Course Code: .....

Centre for Internal Quality Assurance (CIQA)
Krishna Kanta Handiqui State Open University
City Office: Housefed Complex, Guwahati-781006

Learner's Feedback on Course

Dear Learner,

Regarding the course as mentioned above, we would like to know your opinions and comments so as to improve the quality of self learning materials in future. Please respond to the following statements by ticking the number you feel most reflect your opinion. After completion of the additional comments, please detach the page and send/mail the same to us at the address given below.

The Director, Centre for Internal Quality Assurance, KKHandiqui State Open University
Housefed Complex, Dispur, Guwahati-781006
(E-mail id: ciqa@kkhsou.in)

1) Approximately how many hours did you spend for studying the units in the course? ☐☐☐

2) Please give your opinions (by ✓ mark) to the following items based on your reading of the block:

Sl. No.	Statements	Strongly Agree	Agree	Neither Agree nor Disagree	Disagree	Strongly Disagree
I)	The SLMs of the course fulfil the learning objectives set out in the beginning of each unit					
II)	The units of the SLM could satisfy my academic needs and expectations					
III)	The Unit writers have excellent knowledge about the course contents					
IV)	Language and contents of the units were presented at a level which I could readily understand					
V)	Ample opportunity for participation in the activities provided in the units					
VI)	Used enough Illustrations (Diagrams, tables etc.) for conceptual clarity					
VII)	Quality of content is engaging, relevant, and up-to-date					
VIII)	The self check questions are very helpful					
IX)	The Possible/model questions and the answers to check my progress have benefited me a lot					

Additional Comments: *(Please feel free to provide your open comments)*

1) Which aspects of the SLM, according to you, worked well?

.....

.....

2) What sort of changes/improvements do you feel KKHSOU could implement to improve the overall quality of the SLM?

.....

.....

Thank you for taking the time to complete this form.